TECHNICAL REPORT

ISO/TR 14813-3

First edition
2000-12-15

Transport information and control systems — Reference model architecture(s) for the TICS sector —

Part 3: Example elaboration

Systèmes de commande et d'information des transports — Architecture(s) du modèle de référence du secteur TICS —

Partie 3: Élaboration d'exemple

Reference number
ISO/TR 14813-3:2000(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.ch
Web www.iso.ch

Printed in Switzerland

Contents

List of figures

List of tables

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The main task of technical committees is to prepare International Standards, but in exceptional circumstances a technical committee may propose the publication of a Technical Report of one of the following types:

— type 1, when the required support cannot be obtained for the publication of an International Standard, despite repeated efforts;

— type 2, when the subject is still under technical development or where for any other reason there is the future but not immediate possibility of an agreement on an International Standard;

— type 3, when a technical committee has collected data of different kind from that which is normally published as an International Standard ("state of the art", for example).

Technical Reports of types 1 and 2 are subject to review within three years of publication, to decide whether they can be transformed into International Standards. Technical Reports of type 3 do not necessarily have to be reviewed until the data they provide are considered to be no longer valid or useful.

Technical Reports are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

Attention is drawn to the possibility that some of the elements of this part of ISO TR 14813 may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO/TR 14813-3, which is a Technical Report of type 2, was prepared by Technical Committee ISO/TC 204, *Transport information and control systems*.

This document is being issued in the Technical Report (type 2) series of publications (according to subclause G.3.2.2 of Part 1 of the ISO/IEC Directives, 1995) as a “prospective standard for provisional application” in the field of transport information and control systems because there is an urgent need for guidance on how standards in this field should be used to meet an identified need.

This document is not to be regarded as an “International Standard”. It is proposed for provisional application so that information and experience of its use in practice may be gathered. Comments on the content of this document should be sent to the ISO Central Secretariat.

A review of this Technical Report (type 2) will be carried out not later than three years after its publication with the options of: extension for another three years; conversion into an International Standard; or withdrawal.

ISO TR 14813 consists of the following parts, under the general title Transport information and control systems — Reference model architecture(s) for the TICS sector:

— *Part 1: TICS Fundamental Services*: This document presents the definition of 32 TICS fundamental services that are the informational products or services or applications areas provided to a TICS user.

— *Part 2: Core TICS Reference Architecture*: This document describes an abstract object-oriented system architecture based on the TICS Fundamental Services.

— *Part 3: Example Elaboration*: This document refines the Core TICS Reference Architecture (Part 2) with some emphasis on traffic management.

— *Part 4: Reference Model Tutorial*: This document describes the basic terms, graphical representations and modelling views exploited in the object-oriented definition of the architecture development of Parts 2 and 3.

— *Part 5: Requirements for Architecture Description in TICS Standards*: Requirements for Architecture Description in TICS Standards: This document describes the terminology and form to be used when documenting or referencing aspects of architecture description in TICS standards.

— *Part 6: Data Presentation in ASN.1:* This document establishes the use of ASN.1 as the normal syntax notation to be used in standards for the TICS sector and a common message form for such ASN.1 based data elements.

Annex A of this part of ISO/TR 14813 is for information only.

Introduction

TC204/WG1 is a working group whose prime objectives are to provide services to ISO TC204 and its working groups. A specific mission of WG1 is to:

“Provide ISO TC204, its working Groups, related bodies and those involved in the TICS sector, with a reference model of Conceptual Reference Architecture(s) that show the structure and inter-relationships of the sector ...”

It is expected that there may well be more than one single TICS Architecture approach to be considered and documented and that existing architecture approaches will have previously-produced documentation developed according to disparate standards and conventions.

It is also implicit in the work being undertaken by WG1, that working group members will require a clear, well-structured understanding of the work of the following participant groups:

— Other TC 204 Working Groups

— CEN TC 278 Working Groups

— Japanese initiatives

— European Road Transport and Traffic Telematics programs

— US Intelligent Transportation Systems program

— Australian initiatives

— Canadian Initiatives

Full documentation of all possible architectural approaches is obviously not feasible given the high level of resources required to carry this out. Indeed full documentation and description of all possible approaches is undesirable as an item for Standardisation.

A defined and consistent approach is however required to facilitate the specification of architecture requirements to enable a clear view to be developed and presented of the work of each participant group This document is one of a set of WG1 documents intended to respond to stated WG1 objectives regarding the production of a TICS Reference Architecture.

In order to document an architecture, graphical and textual components of a model are required. WG1 has adopted a methodology based on the Unified Modelling Language (UML) for documenting the TICS Reference Architecture. A tutorial on the UML is provided in ISO/TR 14813 Part 4. UML is a visual modelling language for building object-oriented and component-based systems. A commercially available Computer Aided Software Engineering (CASE) tool has been used by WG1 to document the Architecture. While the tool is a commercial product, UML is open and non-proprietary.

Transport information and control systems — Reference model architecture(s) for the TICS sector — Part 3: Example elaboration

1 Scope

The architecture of an information and control system merges hardware and software considerations into a coordinated and integrated system view. The system architecture is a high level abstraction, or model, of the system. A system architecture should embrace both today's applications and the applications that are expected in the future. Architecture begins with the definition of the conceptual services (e.g. Part 1 - TICS Fundamental Services). There are several identifiable stages of system architecture development.

a) Reference architecture

b) Logical architecture

c) Physical architecture

A reference architecture is the first of all architectures. It is a concise generic framework which guides the development of more concrete system architectures. It is large enough that distinct concepts are not merged out of necessity and small enough that it does not become unwieldy.

A most significant example of a reference architecture in information systems is the Reference Model of Open Systems Interconnection (often called the seven layer model) developed by ISO in the 1970's. This model has underpinned the development of all modern computer networks, allowing services such as global networking, of which the prime example is the Internet, to become a reality.

A reference architecture is generic and non-prescriptive and captures the concepts of the system. A logical architecture elaborates the conceptual behaviour, and in so doing it provides more detail about the modularity. A physical architecture is reached when the actual distribution of the system modules is defined, thus leading to important implications for communications.

There is no firm demarcation between a reference architecture and a logical architecture. Thus the essence of behaviour and modularity is present in a reference architecture. The TICS Reference Architecture developed by WG1 shows important inter-relationships that arise in the provision of the services of the sector. However the TICS Reference Architecture is more abstract than, for example, the logical architecture of the US National Architecture.

It is envisioned that the TICS Reference Architecture will be used by the TC204 Working Groups to develop their own logical and physical architectures in a cohesive manner.

Some TICS Fundamental Services are already well developed by the industry, while others are less mature. Therefore the TICS Reference Architecture does not have a uniform granularity across all services. This characteristic is a direct result of the fore mentioned requirement that architecture embrace the applications that are intended in the future. This suggests one of the ways in which the architecture will undergo change in the future.

Architectures may present only static characteristics or both static and dynamic characteristics. Dynamic characteristics may be seen as belonging solely to the design/implementation stages of system development. However by including dynamic characteristics at the reference architecture stage one can gain important insights into the static architecture. Thus two orthogonal views of architecture are presented:

a) static relationship view (class diagram)

b) dynamic interactive view (sequence diagram)

Part 2 develops a Core TICS Reference Architecture. The static scope is determined by deriving the system boundary and the use cases from an analysis of the TICS Fundamental Services (Part 1).

The Core Reference Architecture is a reference for the development of national architectures.

This Part elaborates the core by refinement of the two orthogonal views. The elaboration calls upon domain expertise which would be provided by other TC204 Working Groups in the development of ISO standards, or by national groups developing national architectures and standards.

The Core Reference Architecture is elaborated in Clauses 6 to 8. Clause 5 introduces the elaboration method employed. Clause 6 elaborates the classes. Clause 7 elaborates the sequence diagrams. Clause 8 describes the elaborated packages. Clauses 9 and 10 identify some of the main dependencies between the packages.

Readers should refer to Part 4 (Tutorial) for an introduction to the modelling views used in this Part and the overall methodology.

2 Normative references

The following normative documents contain provisions that, through reference in this text, constitute provisions of this part of ISO/TR 14813. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of ISO/TR 14813 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO/TR 14813-1:1999, *Transport information and control systems – Reference model architecture(s) for the TICS sector – Part 1:TICS fundamental services.*

ISO/TR 14813-2:2000, *Transport information and control systems – Reference model architecture(s) for the TICS sector – Part 2: Core TICS reference architecture.*

ISO/TR 14813-4:2000, *Transport information and control systems – Reference model architecture(s) for the TICS sector – Part 4: Reference model tutorial.*

3 Terms and Definitions

For the purposes of this part of ISO/TR 14813, the following semantic definitions apply.

3.1
The **Unified Modeling Language** (UML) is the industry-standard language for specifying, visualising, constructing, and documenting the artefacts of software systems. It simplifies the complex process of software design, making a "blueprint" for construction.[1]

3.2
A **use case** is a coherent unit of functionality provided by a system or class as manifested by sequences of messages exchanged among the system and one or more outside interactors (called actors) together with actions performed by the system.

[1] http://www.rational.com/uml/

3.3
An **actor** is a role of an object or objects outside of a system that interacts directly with it as part of a coherent work unit (a use case). An Actor element characterises the role played by an outside object; one physical object may play several roles and therefore be modelled by several actors.

3.4
Use case diagrams show elements from the use case model. The use case model represents functionality of a system or a class as manifested to external interactors with the system.

There are several standard relationships among use cases or between actors and use cases.

Communicates - The participation of an actor in a use case. This is the only relationship between actors and use cases.

Uses - A uses relationship from use case A to use case B indicates that an instance of the use case A will also include the behavior as specified by B.

To reinforce the externality of actors a system boundary separates the actor symbols from the use case symbols.

3.5
A **package** is a grouping of model elements. Packages themselves may be nested within other packages. A package may contain both subordinate packages and ordinary model elements. The entire system description can be thought of as a single high-level *system* package with everything else in it. All kinds of UML model elements and diagrams can be organised into packages.

Packages own model elements and model fragments and are the basis for configuration control. Each model element can be directly owned by a single package, so the package hierarchy is a strict tree.

Packages can reference other packages so the usage network is a graph. Relationships drawn between package symbols denote relationships between at least some of the elements in the packages. A dependency relationship implies one or more dependencies among the model elements, in which a change in the targeted element may require a change in the source element.

3.6
A **class** is the descriptor for a set of objects with similar structure, behavior, and relationships. UML provides notation for declaring classes and specifying their properties, as well as using classes in various ways. Classes are declared in class diagrams and used in most other diagrams. UML provides a graphical notation for declaring and using classes, as well as a textual notation for referencing classes within the descriptions of other model elements.

3.7
A **class diagram** is a graph of Classifier elements connected by their various static relationships. (Note that a ''class'' diagram may also contain interfaces, packages, relationships, and even instances, such as objects and links. Perhaps a better name would be ''static structural diagram'' but ''class diagram'' is shorter and well established.)

3.8
An **object** represents a particular instance of a class. It has identity and attribute values. The same notation also represents a role within a collaboration because roles have instance-like characteristics.

3.9
An **operation** is a service that an instance of the class may be requested to perform. It has a name and a list of arguments.

3.10
A binary **association** is an association among exactly two classes (including the possibility of a reflexive association from a class to itself).

3.11
Generalization is the taxonomic relationship between a more general element and a more specific element that is fully consistent with the first element and that adds additional information. It is used for classes, packages, use cases, and other elements.

3.12
A **sequence diagram** represents an **Interaction**, which is a set of messages exchanged among objects within a collaboration to effect a desired operation or result. A sequence diagram shows an interaction arranged in time sequence. In particular, it shows the objects participating in the interaction by their ''lifelines'' and the messages that they exchange arranged in time sequence. It does not show the associations among the objects.

3.13
The **system boundary** depicted in a sequence diagram maps to the same entity in a use case diagram. Thus any interaction emanating or terminating in the system boundary involves an actor.

In the methodology classes are invented for one of three purposes: information, control and interface.

An information class defines objects, which will store data relevant to the operation of the system and the actors and maintain that data with database like services.

A control class defines objects whose primary purpose is to implement the functions of the system.

An interface class defines objects that perform the data presentation and application interfaces for the actors.

3.14
The **architecture boundary** divides the interface classes from those classes, which form the actual architecture, namely the control classes and the information classes.

In the sequence diagrams developed in later clauses there is often an implicit interaction across the system boundary involving an actor. This may be implied whenever a message is initiated or terminated at an interface class.

4 Symbols and Abbreviated Terms

4.1 Use Case diagram

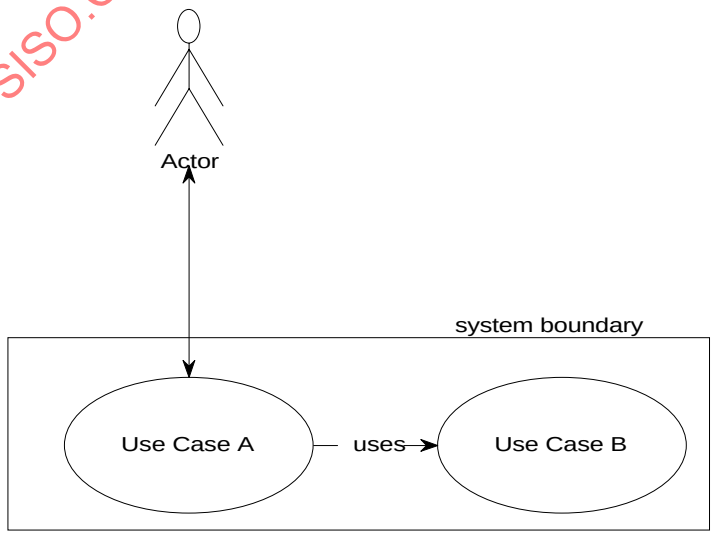

Figure 1 — A use case diagram consisting of two use cases and one actor

4.2 Package Diagram

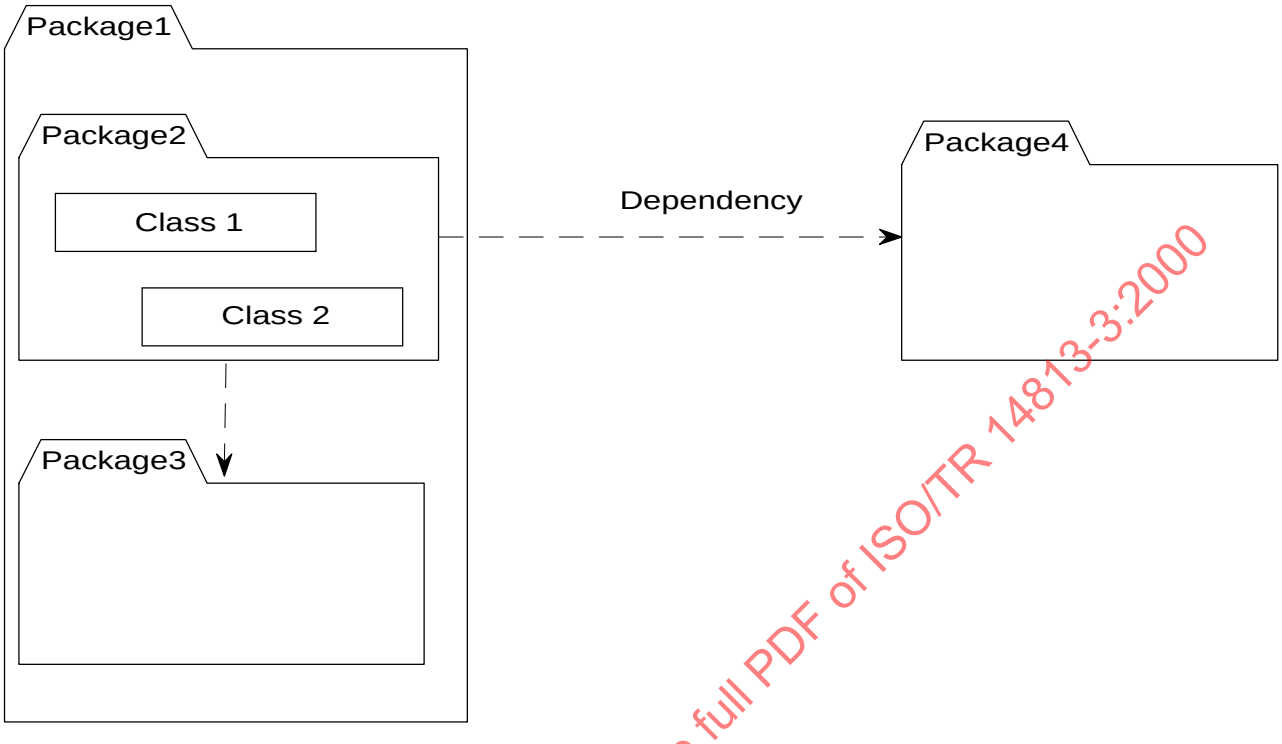

Figure 2 — Package diagram showing nested packages and a dependency relationship

4.3 Class diagram

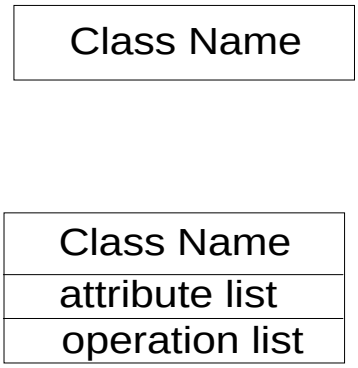

Figure 3 — Class diagrams showing the use of a single name compartment and three compartments

4.4 Association

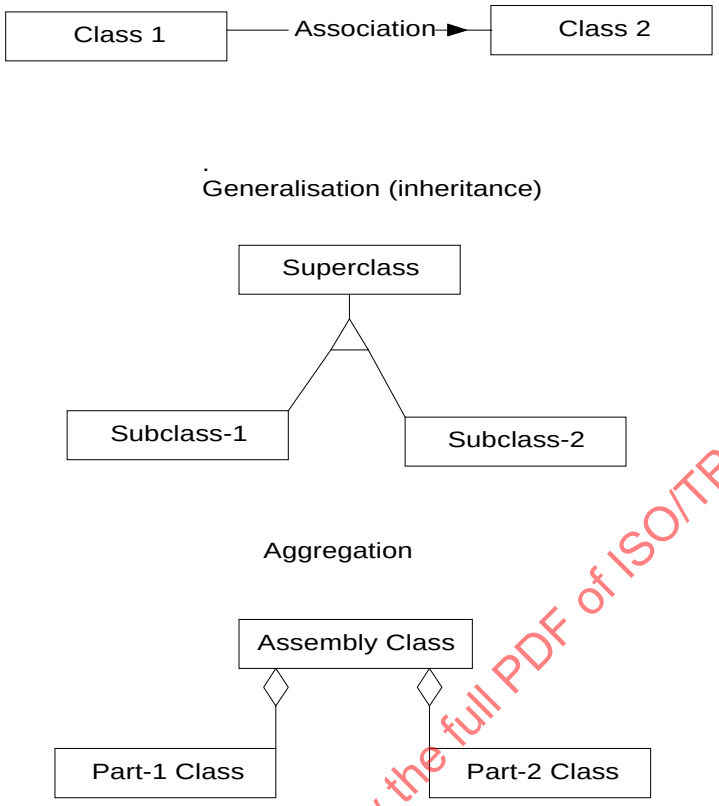

Figure 4 — Common types of class association

4.5 Sequence (interaction) diagram

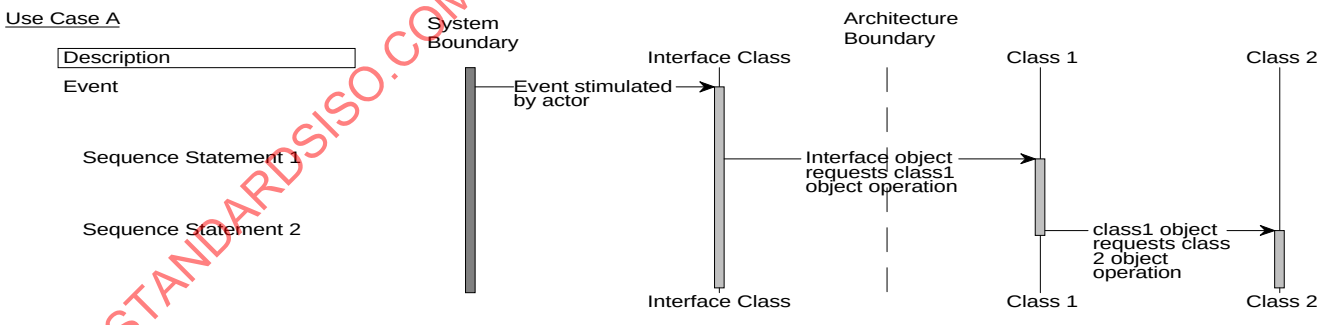

Figure 5 — A sequence diagram consisting of three interactions

5 Elaboration Method

The core Reference Architecture has been defined in ISO/TR 14813 Part 2, using the UML metatypes actor, use case, package and class, as well as sequence diagrams. All the instances of the metatypes in the core architecture are highly abstract. They require a good deal of elaboration before they can be of use in the standards process. That elaboration is now taken a step further.

The elaboration method is presented in Figure 6. There are four stages. While these are ordered in the figure, the whole process is iterative because of the necessity to formulate many new classes and their operations, guided by the principle of reuse.

The first stage is a transformation and refinement of the abstract classes defined in ISO/TR 14813 Part 2. The process is based on an analysis of the class operations identified in the core reference architecture. This analysis leads to some fairly obvious partitions in the light of domain knowledge already documented in the associated use case and the sequence diagram. Each partition is used to define a new control class. Other control classes are identified in the third stage. This first stage is conducted in clauses 6.1 and 6.2.

The second stage would actually be spread throughout the third stage. It involves the formulation of a wide range of information classes that serve to define the parameters of the class operations involved in the object interactions of the new sequence diagrams. In this report the activity is only partially completed. It is focused on traffic management where a large number of information classes are identified from a single reference source.

The third stage is the key to the elaboration and is presented in clause 7. This stage constitutes a major refinement of the sequence diagrams of the core architecture. A sequence diagram is developed for each of the use case identified in the diagrams of the core architecture. Elaborating the logic of each use case results in a significant set of operations being defined for each class. In some cases the parameters of these operations are identified. The process results in the definition of more classes not previously identified in the first two stages.

The fourth stage is a redefinition of the packages based on the expanded set of classes. This is presented in clause 8 for the key classes. Some detailed packaging is defined in 7.2 for classes relevant to Traffic Management.

6 Elaboration of the Classes

The class refinement replaces each abstract class from ISO/TR 14813 Part 2 with a number of classes. These classes fall into the categories:

c) control classes

d) information classes

e) interface classes

Often there is not a clear-cut distinction between control and information classes, but there is usually a dominant role which will be noted. Each class is associated with one of the packages defined in ISO/TR 14813 Part 2.

As in ISO/TR 14813 Part 2 the classes will have operations defined in accordance with the logic of the use case. Here the elaboration of the architecture also includes the definition of the parameters of some of the operations. The abstract level of the parameters means that it is appropriate to define them as information classes associated with the relevant control class. (These information classes are considered to be attributes of the associated control class. Association is the normal means by which complex attributes are implemented in object-oriented software.)

The elaboration will be carried out in several stages. The next clause defines the most important control classes.

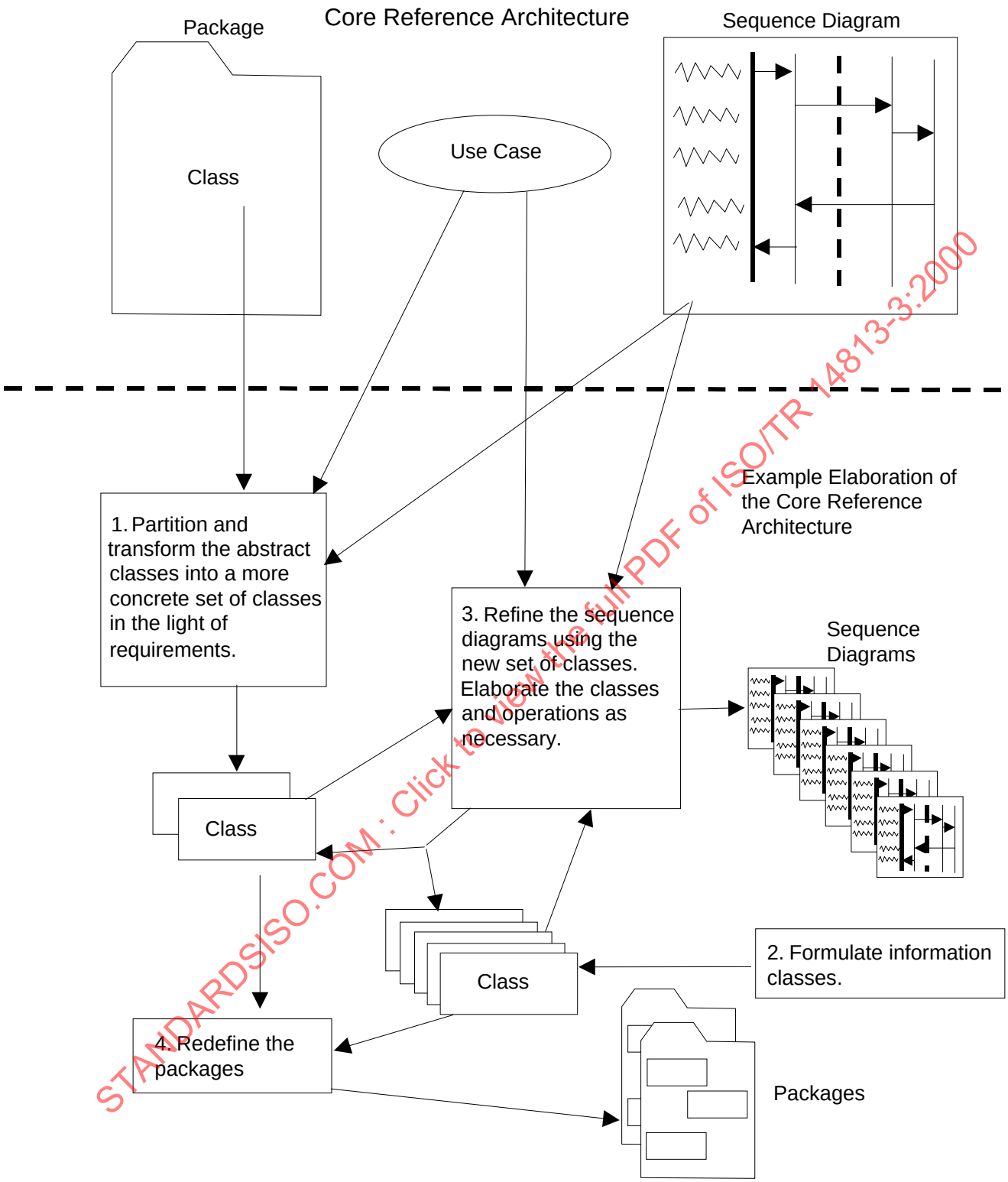

Figure 6 — Steps in the elaboration of the Core Reference Architecture

6.1 Control Classes

This clause infers a number of control classes by systematically grouping the operations of the abstract classes of ISO/TR 14813 Part 2 using general domain knowledge. This systematic grouping is a connection to reality. Each group corresponds to a new control class.

The operations of these new classes will be defined subsequently at a finer granularity, but they will have to deliver the functionality implied by the group of operations derived from the abstract class. Usually the new control classes will have association relationships. The rationale for the identification of each class is presented below.

The reader may wish to refer to ISO/TR 14813 Part 2 for definitions of the abstract operations occurring in the tables below.

6.1.1 Roadway Classes

The operations of the abstract Roadway Class in ISO/TR 14813 Part 2 are classified and grouped in Table 1. This permits the Roadway Class to be replaced in a refinement process by three classes.

Table 1 —Control classes and operations of the Roadway package

Resource	Local Control Group	Roadway Group
accept user credit	measure & local control	access external data
detect and identify vehicle	priority request	access roadway conditions
resource use statistics		compute control actions
vehicle safety and logging		data fusion
		evaluate
		manage demand
		plan construction and maintenance
		predict
		priority request
		support planning
		update geographic data

6.1.1.1 Resource

This class is invented for the purpose of monitoring the use of parts of the roadway network, particularly where commercial vehicle taxes, tolling, and certain safety aspects are involved. A resource object would model a part of the network for which a financial or logging transaction may occur.

6.1.1.2 Local Control Group

This class is invented for the purpose of monitoring and controlling a small or confined part of the roadway such as an intersection or a section of motorway or several such elements which have a geometric relationship. An object of this class might contain one or more Resource objects as shown by the aggregation association in Figure 7.

6.1.1.3 Roadway Group

This class is invented for the purpose of exercising strategic control over the roadway network. An object of this class might contain some geographically extensive Resource objects and a number of Local Control Group objects, the aggregation being defined in strategic design steps. These associations are shown in Figure 7.

There will be additional operations defined for these control classes which allow their cooperation in line with defined associations. Many information classes will be defined and associated with the Roadway Group class to serve the control operations of the classes in Figure 7.

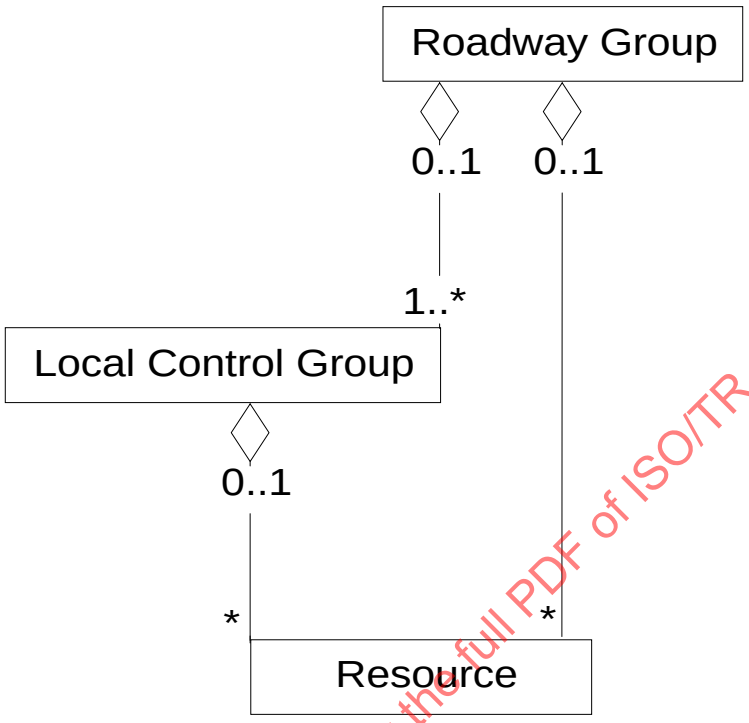

Figure 7 — Control class diagram of the Roadway package

6.1.2 Transport Classes

The operations of the abstract Transport Class in ISO/TR 14813 Part 2 are classified and grouped in Table 2. This permits the Transport Class to be replaced in a refinement process by six classes.

6.1.2.1 Route

This class is invented to model the geographic characteristics for a route. It is specialised into two important sub-classes as shown in Figure 8.

6.1.2.2 O-D Route

This sub-class of Route is invented for the purpose of describing routes for important representative trips. Operations of this class maintain data about services, prevailing conditions and performance, thus it has a role as an information class.

6.1.2.3 Public Transport Route

This sub-class of Route is invented for the purpose of defining and operating the routes of public transport services. The operations also maintain information about stops and interchanges on the roadway network of a public transport service.

Table 2 — Control classes and operations of the Transport package

O-D Route	Public Transport Route	Schedule	Journey	Commercial Vehicle	Forwarder
evaluate post predictions	access running data create & maintain routes & schedules process operational data response request service change request	ad hoc inquiry reservation service	establish journey conditions and options plan journey schedule journey	access incidents monitor vehicle & cargo recognise vehicle	select forwarder

6.1.2.4 Schedule

This class is invented to model all types of public transport schedule. The operations relate to time-tabled and demand responsive public transport services. They maintain time-tables, operational data, and demand responsive vehicle itineraries and dispatching. Every Public Transport Route object has a Schedule object associated with it.

6.1.2.5 Journey

This class is invented to serve route and schedule planning for a transportable item. The operations develop the overall plan for Travellers (i.e. Passengers and all types of Drivers).

It is also an information class for the total movement associated with a journey request. The request might be for a goods item to be taken from Consignor to Consignee, or for a Traveller planning a multi-leg Journey.

Every Journey object has one or more O-D Route objects associated with it, corresponding to the number of legs or trips whch make up the Journey object.

6.1.2.6 Commercial Vehicle

This sub-class of Vehicle (see 6.1.3) is invented for the purpose of controlling commercial vehicles on route and interacting with classes in the Roadway package.

A Commercial Vehicle object which has been assigned to a tour will be associated with a Journey object.

6.1.2.7 Forwarder

This class is invented to model some functions of freight movement which are inside TICS.

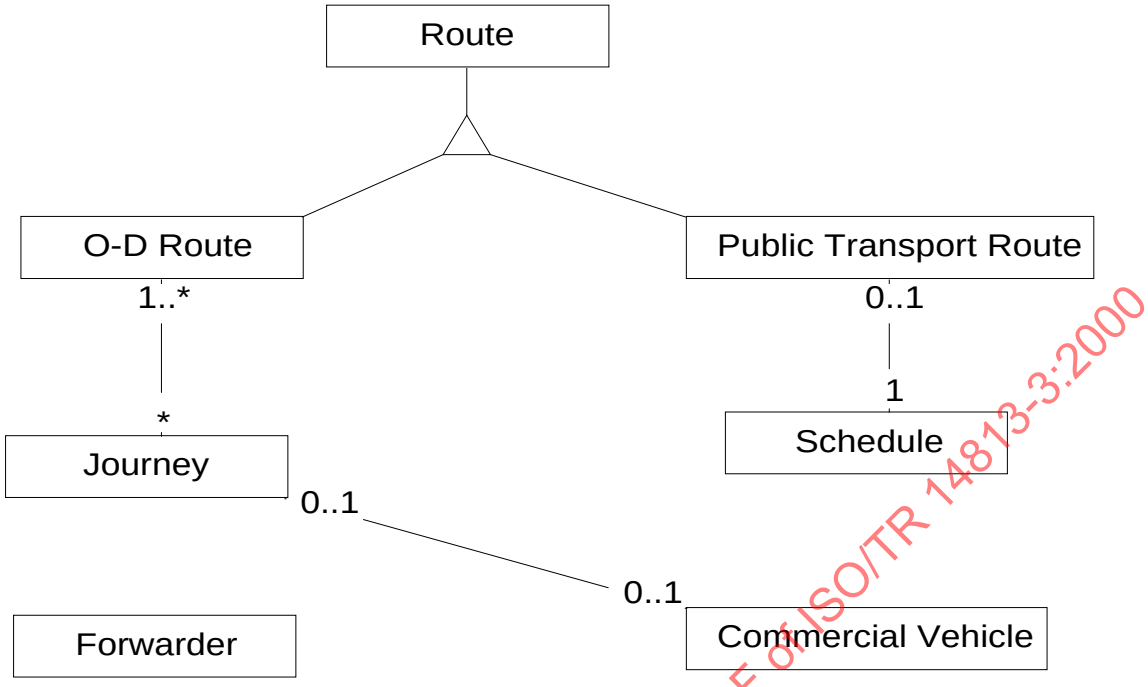

Figure 8 — Control class diagram of the Transport package

6.1.3 Vehicles Classes

The operations of the abstract Vehicle Class in ISO/TR 14813 Part 2 are grouped and retained under a single class in Table 3.

6.1.3.1 Vehicle

This class is invented to model the control of a vehicle for safety and basic operational purposes.
There are special classes of vehicle which inherit these characteristics.

Table 3 — Control classes and operations of the Vehicle package

Vehicle
control vehicle
data fusion
monitor context
support planning

6.1.4 Event Classes

The operations of the abstract Events Class in ISO/TR 14813 Part 2 are classified and grouped in Table 4. This permits the Events Class to be replaced in a refinement process by three classes. (See Figure 9)

Table 4 — Control classes and operations of the Events package

Incident	Emergency	Violation
access incidents	develop strategy	process violations
predicted incidents	process emergency	
suppport planning	request emergency response	
update incidents & emergencies	update incidents & emergencies	
evaluate		

6.1.4.1 Incident

This class is invented to control the process of predicting and handling incidents.

6.1.4.2 Emergency

This class is invented to control the process of detecting and handling an emergency. Thus an emergency object may be associated with an incident object, and vice versa.

6.1.4.3 Violation

This class is invented to control the processing of all violations of regulations. In the case of a serious violation a Violation object may be associated with an Emergency object.

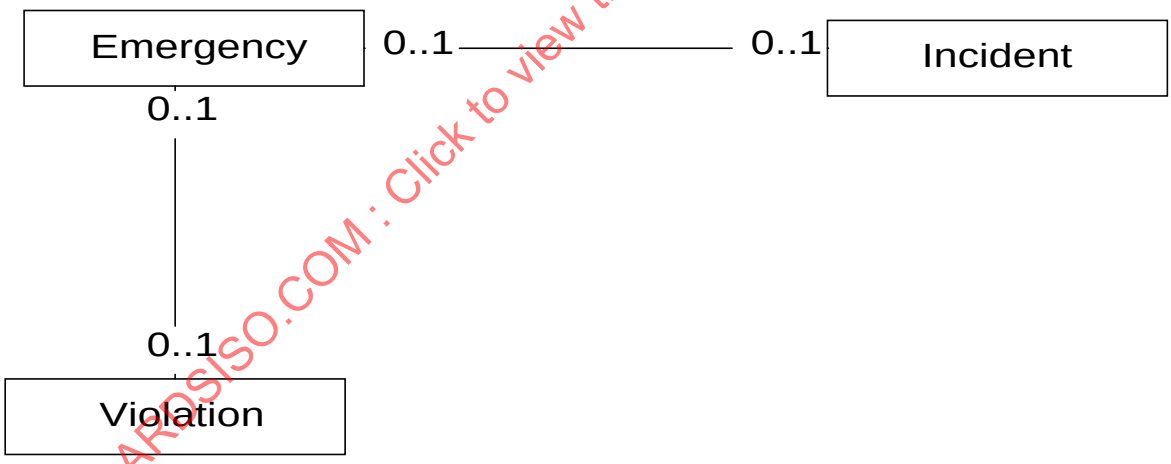

Figure 9 — Control class diagram of the Events package

6.1.5 Payment Classes

The operations of the abstract Payment Class in ISO/TR 14813 Part 2 are classified and grouped in Table 5. This permits the Payment Class to be replaced in a refinement process by three classes.

Table 5 — Control classes and operations of the Payment package

Billing Record	Tariff	Yellow Pages
compute charges	declare tariff	maintain yellow pages
create billing record	response request	
make payment	support planning	

6.1.5.1 Billing Record

This class is invented to support the initialisation and completion of a Resource use, and any payment. A Billing Record object may be associated with a Vehicle object in the case of tolling transactions.

6.1.5.2 Tariff

This class is invented to support the calculation of charges for Resource usage and variations which may be required. A Resource object is associated with a Tariff object.

6.1.5.3 Yellow Pages

This class is invented to maintain the wider information about a general service not held in the Tariff class.

6.2 Interface Classes

The operations of the interface classes are grouped at the same level of abstraction as in ISO/TR 14813 Part 2. They are shown in Tables 6 to 9. The main significance of the interface classes at this stage of architecture elaboration is in the definition of the system boundary and the architecture boundary.

6.2.1 Operating Interface

This interface class generalises all the different types of operating interface. The operations perform data display, support Operator interaction and interaction with a range of external operational systems.

Table 6 — Operating Interface class and operations

Operating Interface
action request
conformance check
request credit issuer payment
reservation request

6.2.2 Travel Terminal

This interface class generalises all the street kiosk interfaces as well as internet and other interfaces. The operations perform data display and support Traveller interaction.

Table 7 — Travel Terminal class and operations

Travel Terminal
accept cash/credit
advertise
debit
payment request
pre-payment
publish traveller information
sense security violation or alert

6.2.3 Vehicle Interface

This sub-class of Travel Terminal is part of the Vehicle package and generalises in-vehicle interfaces. The operations perform data capture and display, other user communications, and support Driver and Passenger interaction.

Table 8 — Vehicle Interface class and operations

Vehicle Interface
access payment means
access vehicle & passenger data
command driver
deploy vehicle and instruct driver
interrogate fleet
notify driver
track vehicle & status

6.2.4 Roadside Peripheral

This interface class is part of the Roadway package. The objects interface signs, some types of sensors and communications beacons. The operations support information flow between Drivers and the Roadway control classes.

Table 9 — Roadside Peripheral class and operations

Roadside Peripheral
notify vulnerable
update control

6.3 Information Classes

With each step in the elaboration of the architecture there is a more rapid growth in the number of information classes than the other types. This is the nature of the large scale information and control system. Control is based on a large amount of real time data, which is reduced to important parameters for key operations. To serve historical, real time and transactional purposes the information classes and their associations constitute schemas for notional object databases.

As noted at the beginning of clause 5, the level of abstraction of an architecture means that the parameters of the class operations are themselves defined as information classes. These information classes are associated with other classes that generate and receive the messages of interest in the architecture.

The definition of the information classes calls for more detailed TICS domain knowledge. In TC204 this domain knowledge will be supplied by the various working groups.

Objects of an information class interact in the standard manner, that is through the operations defined in their interface. These operations maintain data stored in the attributes of the object.

In architecture every information class has the same implicit set of generic operations:

— create(…)

— update(…)

— retrieve(…)

— delete(…)

The difference between each class is the set of attributes and parameters that the operations manipulate. When they serve as parameters in the operations of control classes of the architecture, information objects are always used in their entirety.

6.3.1 Information Classes for Traffic Management and Traveller Information

The draft standard ISO 14827 **Message Format and Information Contexts for Traffic Management and Information Centres (TMIC)**[2)] has recently been developed by TC204/WG9. The data definitions in draft ISO 14827 are not data dictionary definitions, rather they are informal definitions of complex data elements. Most of these data elements map to parameters of the operations in the sequence diagrams for the use cases in Traffic Management and Traveller Information.

In elaborating the architecture a uniform approach was adopted in the use of the ISO 14827 data elements. Each data element definition maps to an information class definition, and the text in ISO14827 is used unchanged. Since that document does not use an object-oriented model there is some inconsistency in terminology. For example, ISO 14827 often refers to a “list of objects”; this should be read as the set of objects belonging to a class. A dictionary of the information classes so derived is given in Annex A.

There are 360 data elements in draft ISO 14827, identified in a naming tree. The root structure of this naming tree has been defined as an aggregate of nine information classes in Figure 10. An information class is associated with each top level data element of ISO 14827, using the name in the ISO 14827 naming tree as part of the class name.

The digits in the leading part of the class name are those used in ISO 14827 to indicate the position in the naming tree. They are included because they are a convenient abbreviation for the information class names where they occur as parameters in the sequence diagrams.

2) The classes derived here are based on draft N173H, 20-01-98.

The remaining information classes were defined in nine hierarchies shown in Figures 11 to 19. The methodology was to first view the leaf nodes (data elements) of the naming tree as candidate information classes. Consolidation of these classes was performed as aggregations to the next level, according to the graph of the naming tree. In order to avoid clutter the leaf information classes are not shown in the Figures 11 to 19. However the parameters of the operations in the sequence diagrams of clause 7 of this report will usually be the aggregate classes.

There are a small number of exceptions where an aggregation association was not appropriate.

In Figure 11 "[1---] infrastructureMap" class is specialised into three subclasses. The ISO 14827 definitions prompting these associations are:

— [1---] infrastructureMap: These are maps of the static physical network.

— [11--] generalMap: This base map describes all relevant geographic features of the region.

— [12--] transportMap: These are facilities operated by the relevant service provider.

— [13--] roadMap: No definition.

In Figure 12 "[21--] vehicles" class is specialised into two sub-classes and it has two other classes related in standard associations. The ISO 14827 definitions prompting these associations are:

— [211-] individualVehilceID: No definition – clearly a vehicle attribute.

— [212] governmentID: This information class shall consist of the vehicles (objects) that an agency wishes to have identified in the manner indicated.

— [213-] policeListing: This information class shall consist of the vehicles (objects) that an agency wishes to have identified in the manner indicated.

— [214-] vehicleCondition: No definition – clearly a vehicle attribute.

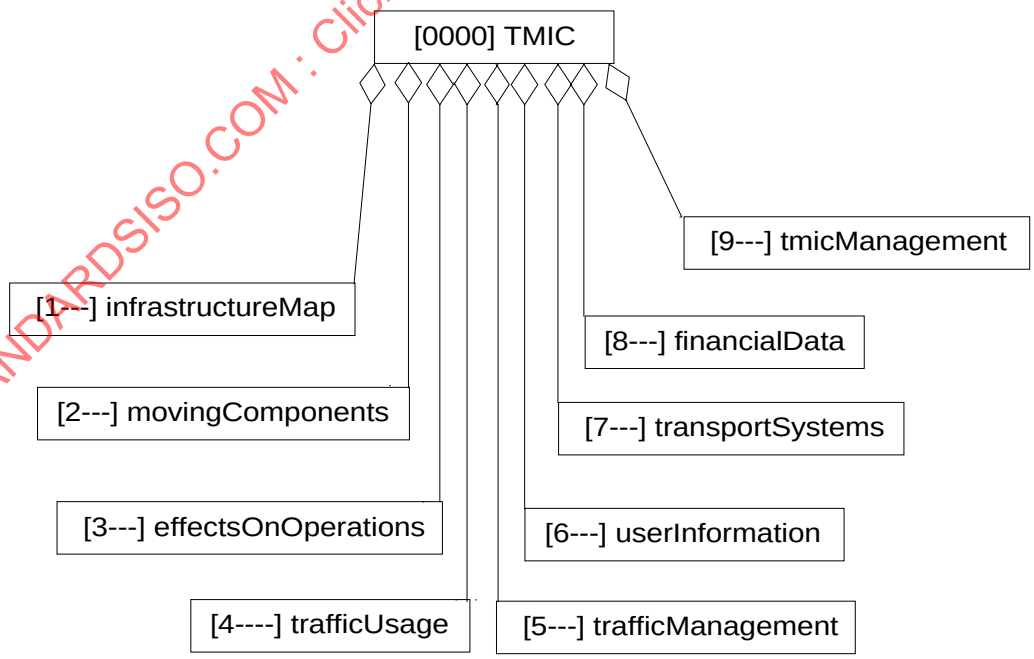

Figure 10 — Top level aggregation of Information classes derived from the 14827 naming tree

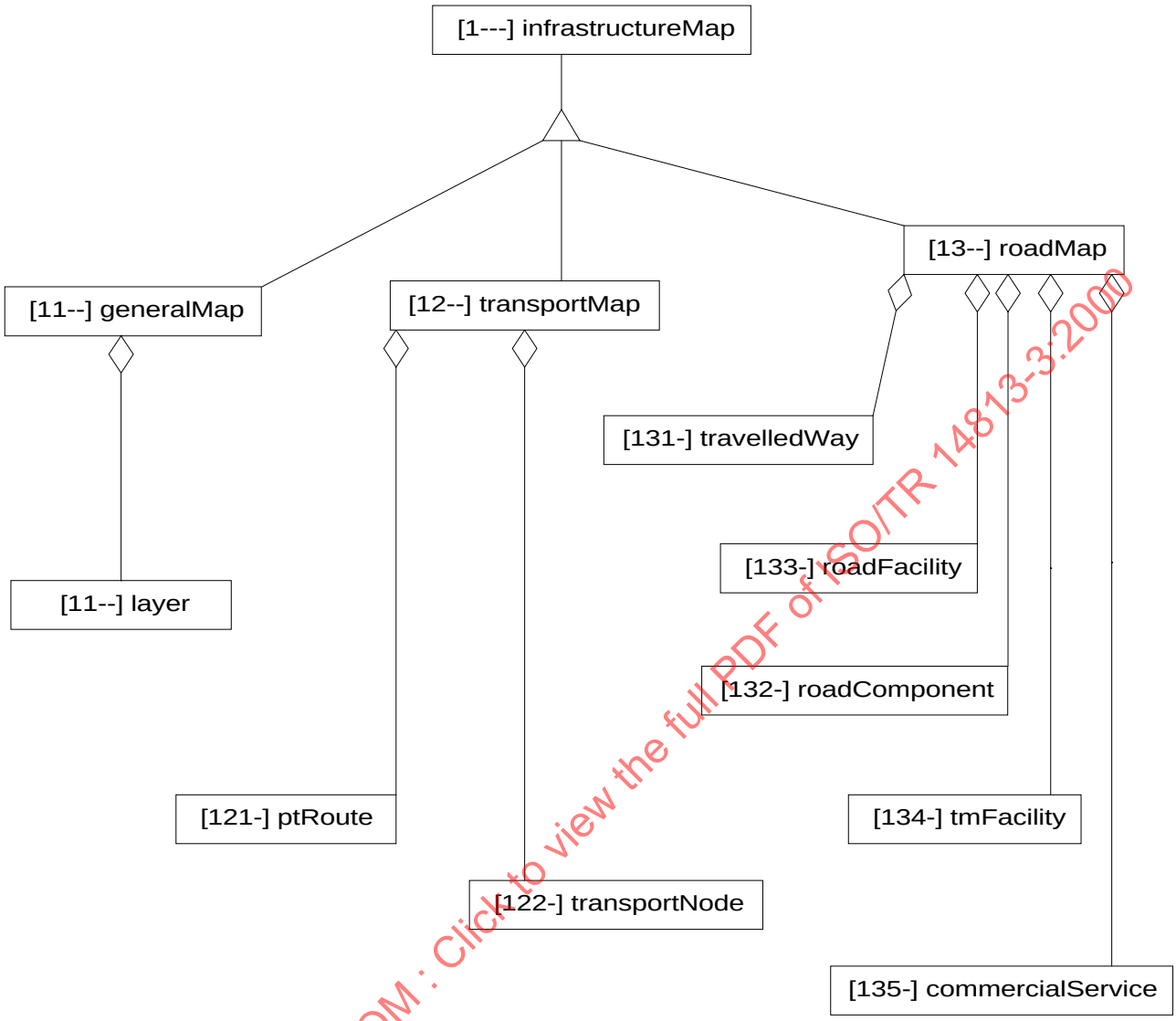

Figure 11 — Information classes in the aggregation of infrastructureMap

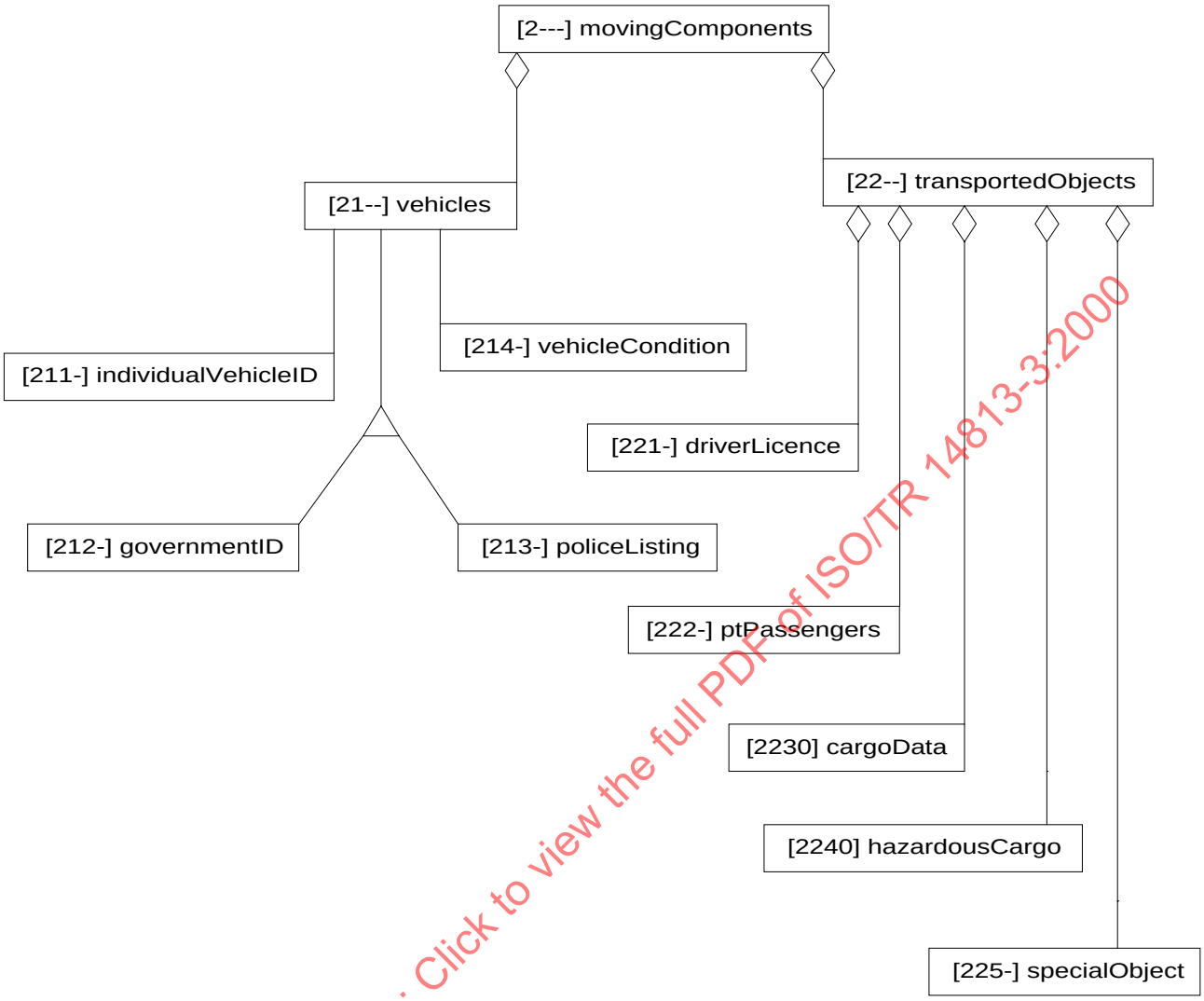

Figure 12 — Information classes in the aggregation of movingComponents

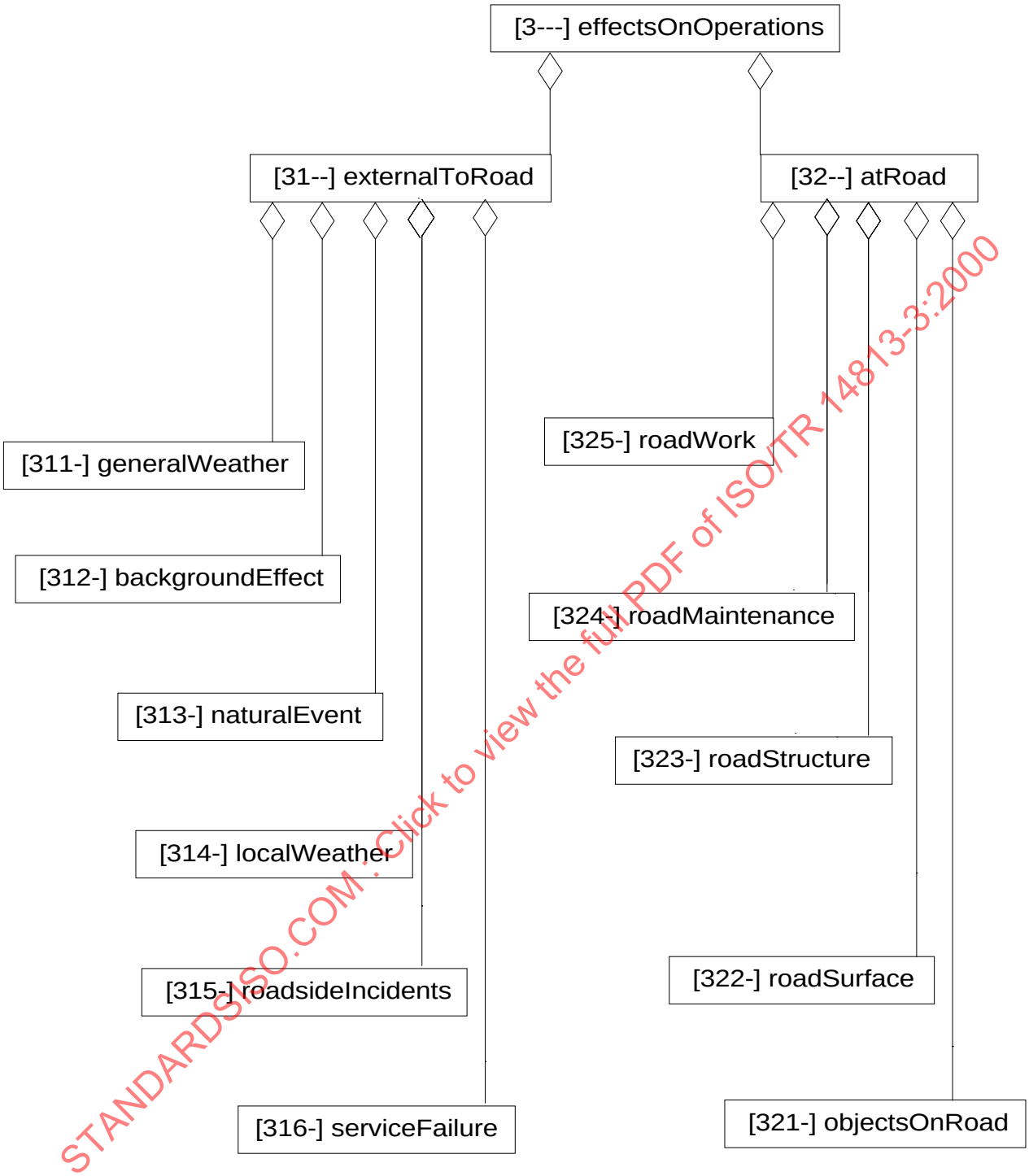

Figure 13 — Information classes in the aggregation of effectsOnOperations

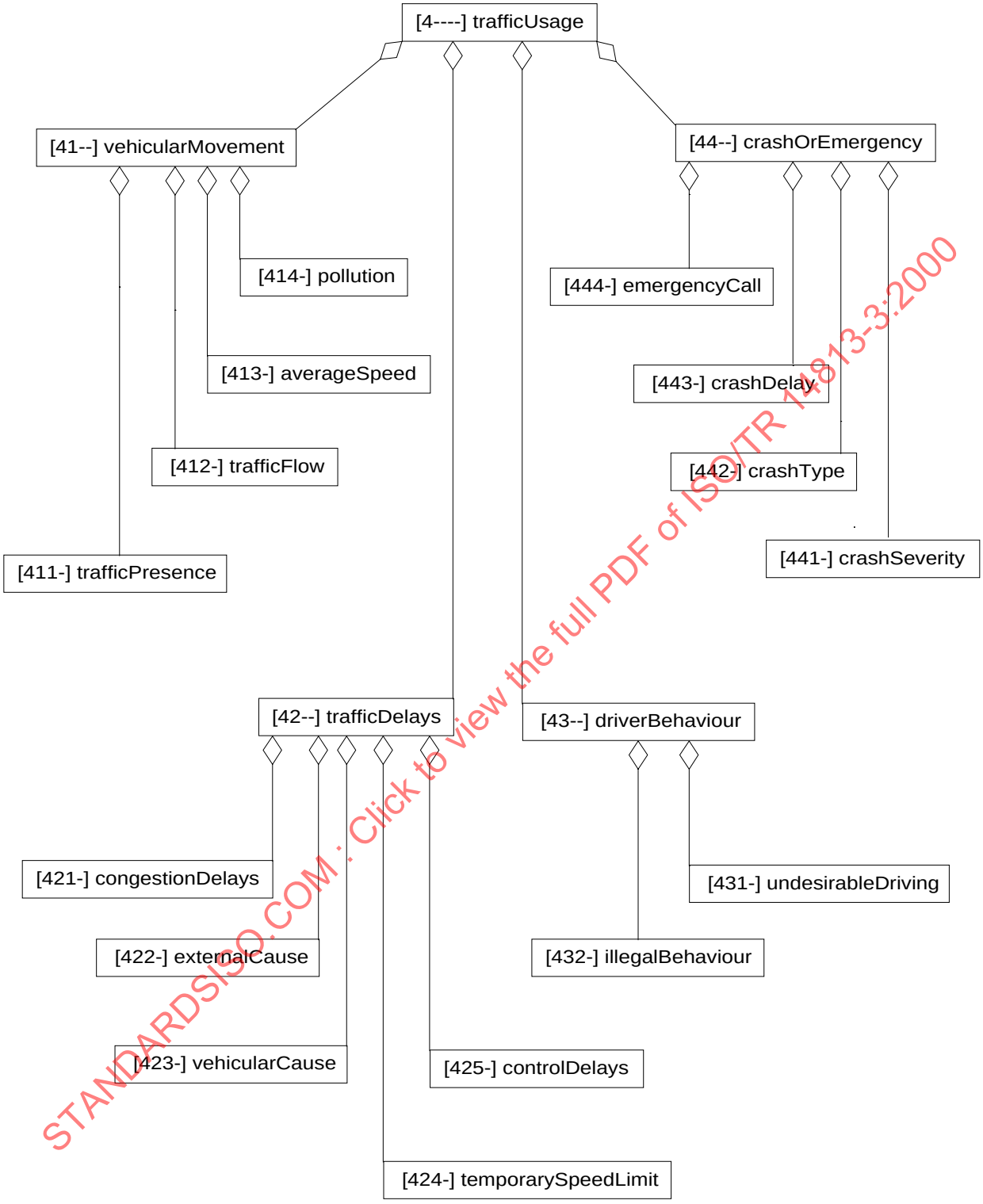

Figure 14 — Information classes in the aggregation of trafficUsage

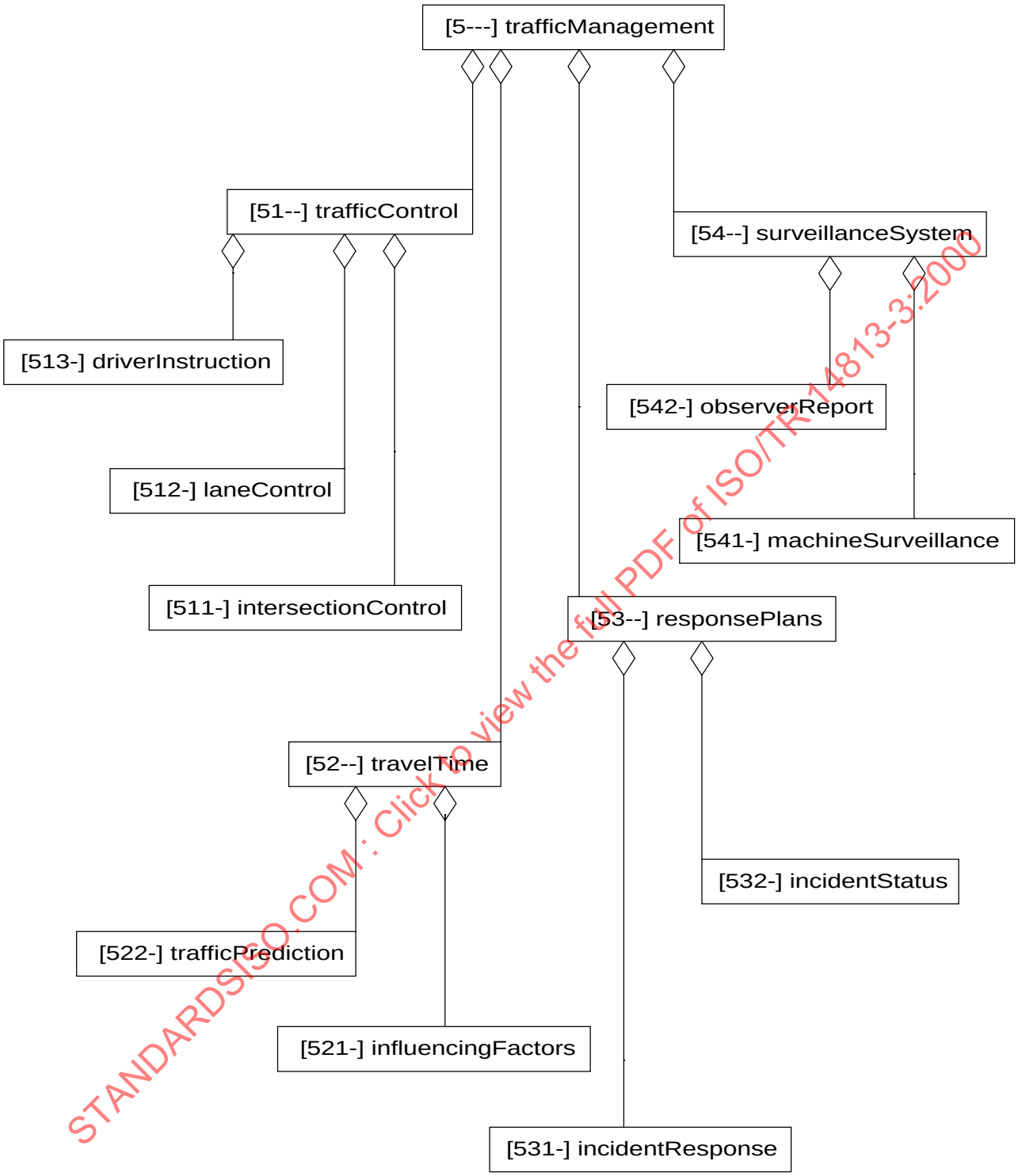

Figure 15 — Information classes in the aggregation of trafficManagement

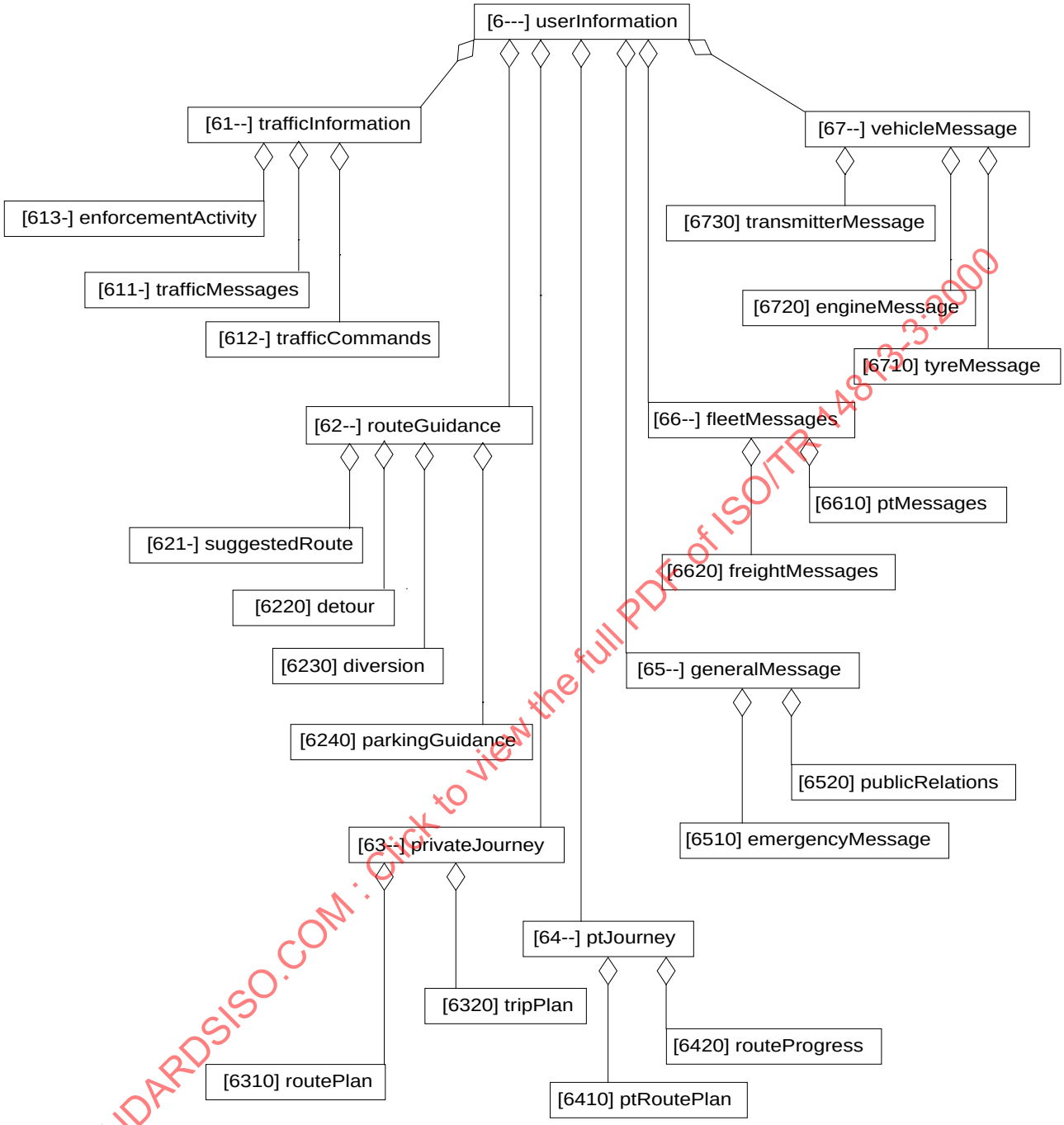

Figure 16 — Information classes in the aggregation of userInformation

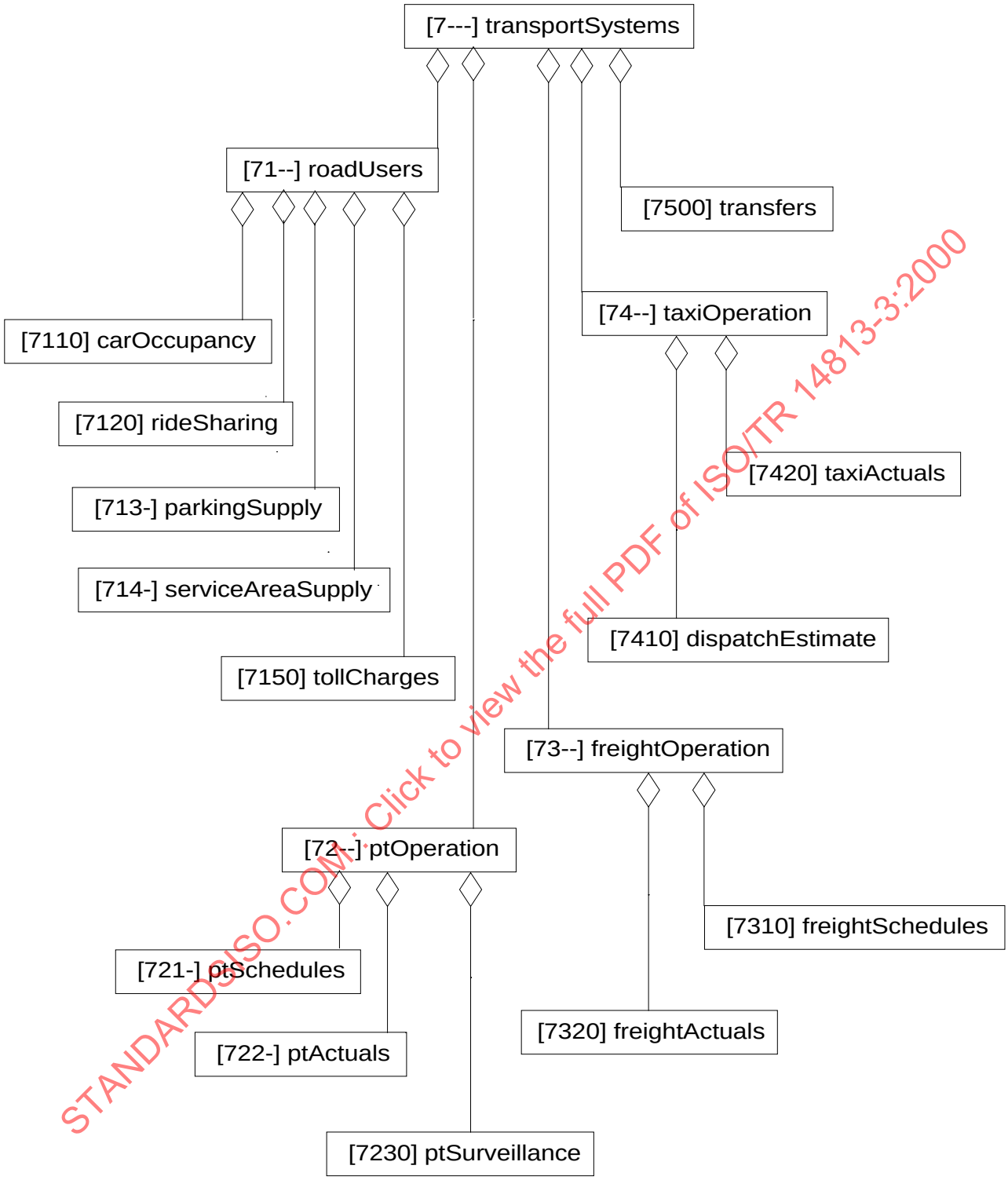

Figure 17 — Information classes in the aggregation of transportSystems

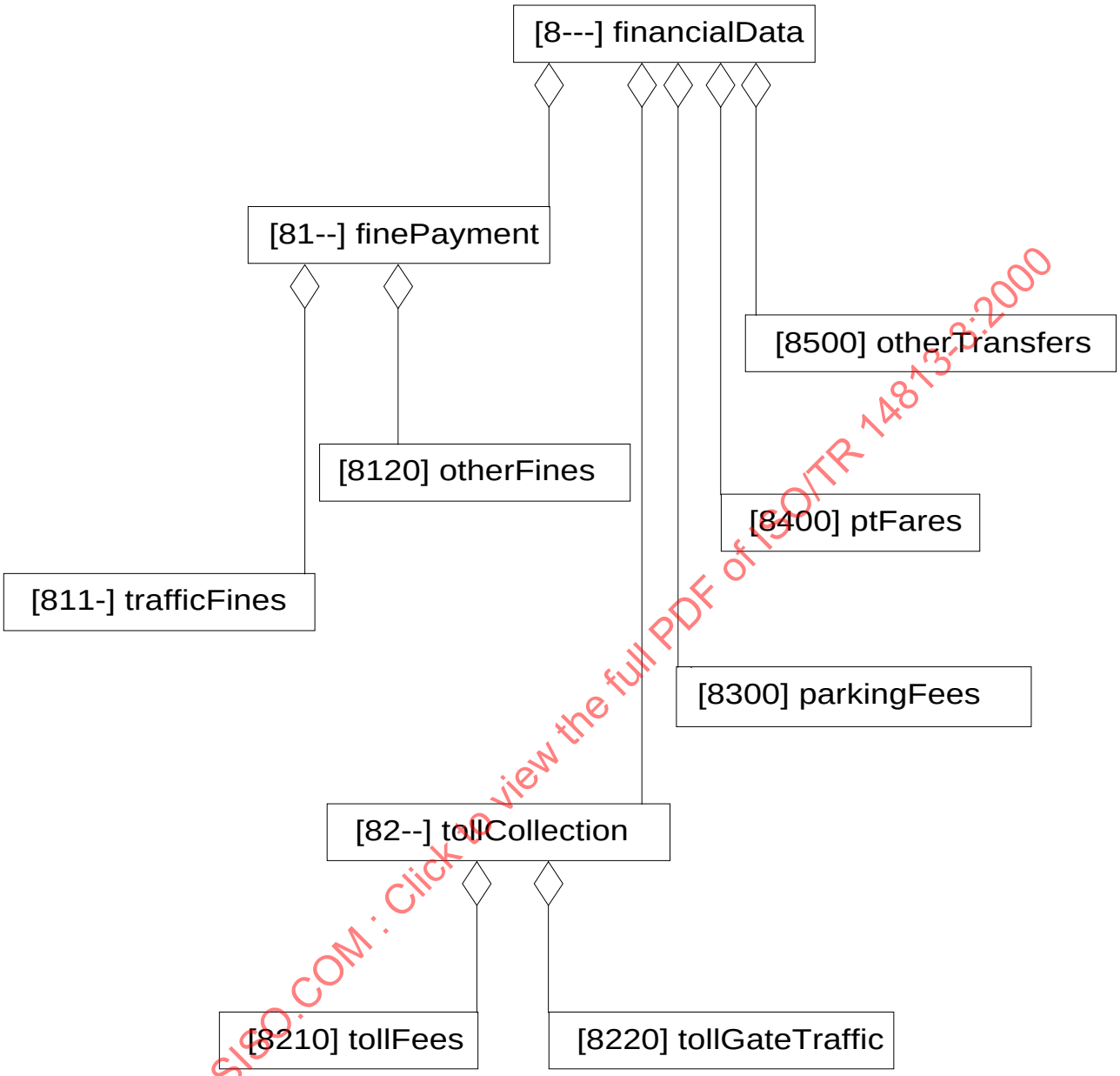

Figure 18 — Information classes in the aggregation of financialData

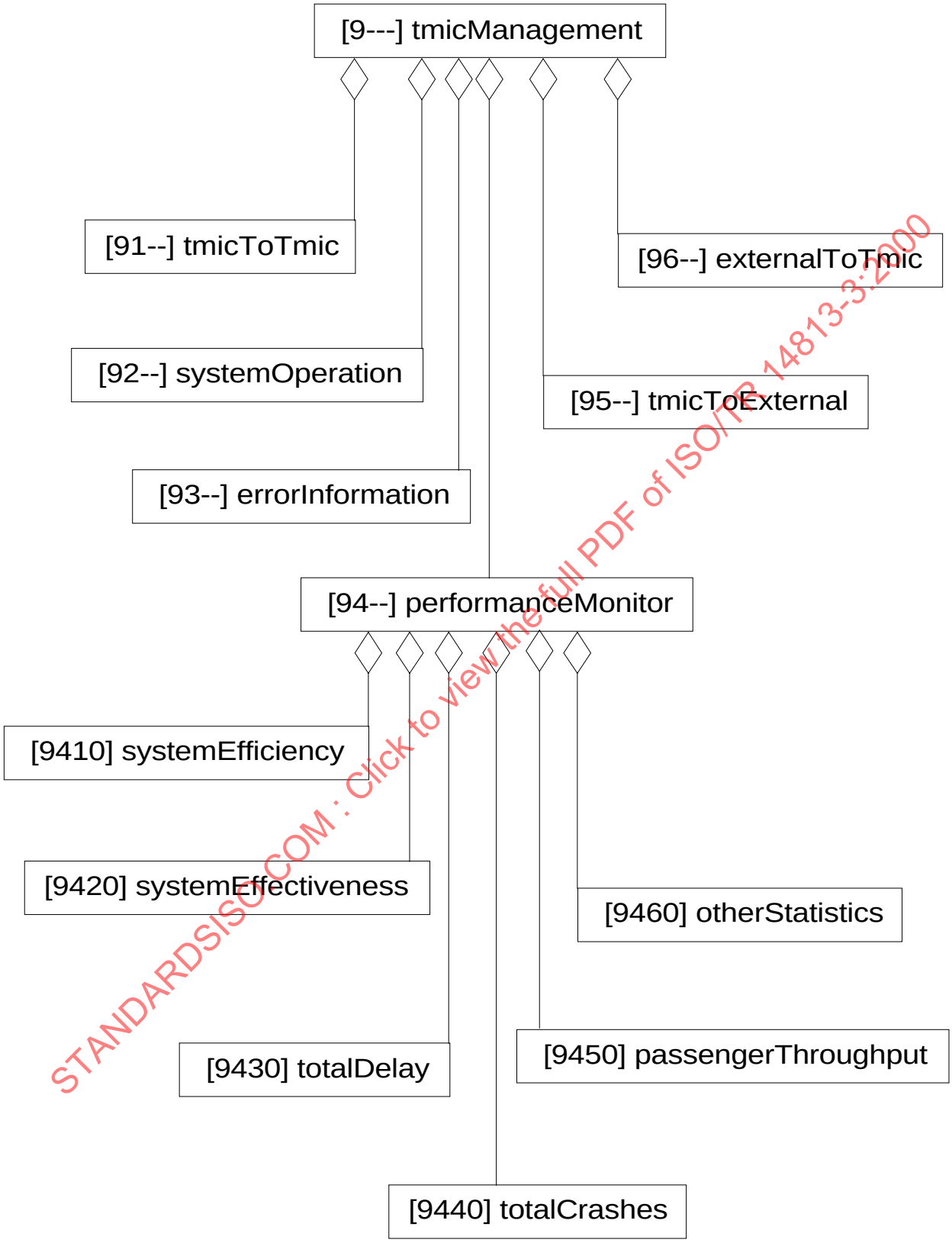

Figure 19 — Information classes in the aggregation of tmicManagement

7 Elaboration of the Sequence Diagrams

This clause presents the results of applying the elaboration method described in clause 5. The primary inputs are the sequence diagrams of the core architecture in ISO/TR 14813 Part 2 and the new classes derived in clause 6 of this document. The object interactions for each use case defined in the abstract sequence diagrams of ISO/TR 14813 Part 2 clause 8 are expanded by mapping each interaction to the expanded set of control classes. In this process the number of object interactions is increased and the class operations defined in ISO/TR 14813 Part 2 are replaced with a finer grained set of operations for the new control classes.

Since sequence diagrams document the logic of the use case, there is a one to one mapping with each use case of the core reference architecture in ISO/TR 14813 Part 2. All the classes involved in a sequence diagram are first identified including some new classes which will be explained as they arise. Any class that provides an operation for an object interaction in a sequence diagram is called a *key class* in this report. Thus key classes are distinguished from classes which only occur as parameters in an object interaction of a sequence diagram.

Within the sequence diagram, each object interaction is represented by a labelled directed line between two objects that are identified by their class name (see Figure 20). The number of pairings of objects is not specified, that is the mapping of an interaction to source and target objects may be one to one, one to many, many to one, or many to many. Also, many occurrences of the same sequence may occur in parallel. The label on the interaction is the name of the operation of the target class which is executed in order to perform that part of a use case transaction. The parameters of the operation (which are transmitted between objects, and which may be input parameters or output parameters or both) are identified in parenthesis after the operation name. In this elaboration the only parameters identified are those defined in 6.3. Thus no parameter list can be regarded as complete, particularly where there are no parameters listed.

Sometimes a parameter of an operation does not map exactly to an information class of 6.3. In these cases, only part of an aggregate object conforming to the specified information class may be relevant as a parameter. If, for example, [41--] is a parameter, this means that there could be one or more of the objects covered by that class aggregation, i.e. [411-] trafficPresence (and recursively any sub-aggregate objects as defined in Annex A), [412-] trafficFlow, [413-] averageSpeed, [414-] pollution. In order to avoid clutter only the digits occurring in the class name are used to identify parameters on the sequence diagram. These digits constitute a unique identifier.

The classes defined in 6.3 have been developed for applications in traffic management and information centres, consequently they are most relevant to the Traffic Management sequence diagrams (7.2) and partly to Traveller Information (7.1). Therefore the development of the sequence diagrams for Traffic Management (7.2) is treated in more detail than that of the other clauses. The object interactions defined for the sequence diagrams in other clauses are largely without parameters.

7.1 Traveller Information

7.1.1 Pre-journey Information

The abstract operations for Pre-journey Information in ISO/TR 14813 Part 2 are provided by the Transport Class. This leads to the following key classes defined in previous clauses.

— O-D Route

— Journey

The following new control and information classes are required:

— Journey Schedule: This class is invented to maintain the mode segments and service times for the trips generated for a journey. The class is associated with the Journey class.

— Public Transport Schedule: A control class which supports the planning and operational functions of a fixed route public transport system.

— Trip: This class is invented to control journey planning and execution. Journeys (for Travellers, Goods Items or Commercial Vehicle Tours) are made up of one or more single mode Trips, thus there is an aggregate association between the Trip and Journey classes. The operations of this class access route information, choose the most suitable route and provide dynamic information.

The sequence diagram for a journey planning exercise is shown in Figure 20. The first stage is to iteratively validate the Traveller's journey request:

— Travel Terminal**.**initiate journey request ([641-])

in the light of feedback information got by comparing requirements with current conditions recorded by objects of the O-D Route class which in turn interact with objects of the Public Transport Schedule class:

— O-D Route.access current conditions ([611-],[72--],[7120],[7131])

— Public Transport Schedule.access public transport running data ([72--])

The main stage is an iteration whereby the validated Journey specification is first defined:

— Journey**.**create ([641-])

Some transactions of Route Guidance and Navigation (Figure 21 at line 2.2) are then invoked which develop the individual Trip and Journey Schedule objects associated with the Journey object. The details of the journey plan are then returned to the Traveller through the nested sequence of operations:

— Journey**.**iterate over trips (…)

— Trip**.**access next segment (…)

— Trip**.**access route ([641-])

— Journey Schedule**.**access schedule ([72--])

If the Traveller commits to this plan the transactions of Journey Payment (7.1.4) are invoked otherwise the journey request is updated and the iteration repeated.

The final transaction is to update the travel statistics:

— O-D Route**.**update travel date([62--])

7.1.2 Route Guidance and Navigation

The abstract operations for Route Guidance and Navigation in ISO/TR 14813 Part 2 are provided by the Transport Class, the Roadway Class and the Event Class. This leads to the following key classes defined in previous clauses.

— O-D Route

— Journey

— Trip

— Public Transport Schedule

— Roadway Group

— Incident

The following new information class is required.

— Probe: This information class is invented to generate resource use statistics from individual travel records which are provided by the Resource control class.

The sequence diagram is shown in Figure 21. The first part is a maintenance iteration for geographic data:

— O-D Route.update map data ([131-])

— Journey.update map data ([12--])

for traffic data:

— O-D Route.update effects data ([3---])

— Roadway Group.access prediction data ([522-])

— Probe.access route travel time data ([131-],[52--])

— Roadway Group.access traffic data ([52--])

— Incident.access predicted incidents ([5213],[5214])

and public transport data:

— Public Transport Schedule.access public transport running data ([72--])

Related O-D Route objects exchange such data:

— O-D Route.exchange data(…)

and data is prepared for application:

— O-D Route.generate advice(…)

This supports any routine enquiry by Travellers:

— O-D Route.access advisory data ([611-],[7---])

The remainder of the sequence diagram is an iteration for servicing various types of planning, guidance or navigation application. If the request is from a vehicle the journey is defined:

— Journey.create([63--])

The Journey (for whatever payload) is then analysed into component trips and the navigation or guidance data generated:

— Trip.create ([631-],[641-])

— O-D Route.access route conditions([631-],[641-],[52--],[7---])

— Trip.generate route and guidance (…)

If the application is for navigation this is accessed as required:

— Trip.access route guidance ([62--])

If the application is for journey planning the transactions of Journey Schedule (7.1.3) are invoked.

7.1.3 Journey Schedule

The abstract operations for Journey Schedule in ISO/TR 14813 Part 2 are provided by the Transport Class. This leads to the following key classes defined in previous clauses.

— Journey

— Trip

— Journey Schedule

The sequence diagram for each journey planning exercise is shown in Figure 22. The first stage is to initialise a schedule:

— Journey Schedule.create (…)

The main iteration is repeated for each trip:

— Trip.access next segment (…)

For those trips which require reservations the action is taken. The Demand Responsive Public Transport transactions (7.5.3) are invoked if necessary. After each actor interaction the confirmed schedule is recorded:

— Journey Schedule.update time confirmation and price (…)

Once all trips have been processed the complete journey is checked:

— Journey Schedule.validate (…)

7.1.4 Journey Payment

The abstract operations for Journey Payment in ISO/TR 14813 Part 2 are provided by Transport Class and the Payment Class. This leads to the following key classes defined in previous clauses.

— Journey

— Journey Schedule

— Billing Record

The sequence diagram for one journey planning exercise is shown in Figure 23. The first stage is to capture the payment means and commit to payment:

— Travel Terminal.read payment card/tag ([2214])

— Journey.commit ([2214])

The main logic is an iteration whereby each trip in the journey is paid for by invoking the transactions of Payment Transaction (Figure 59 at line 4.4.1):

— Journey Schedule.access segment and cost (…)

— Billing Record.create ([2214])

The final action is to give confirmation:

— Travel Terminal.confirm(…)

7.1.5 On-trip Traveller Information

The abstract operations for On-trip Traveller Information in ISO/TR 14813 Part 2 are provided by the Payment Class. This leads to the following key classes defined in previous clauses.

— Yellow Pages Service

— Billing Record

The sequence diagram for the set of transactions is shown in Figure 24. The first part of the sequence is devoted to the maintenance of information services:

— Yellow Pages Service**.**update general information (…)

— Travel Terminal**.**provide yp advertisement (…)

The second part is devoted to the maintenance of yellow pages services:

— Yellow Pages Service**.**update construct yp service (…)

— Yellow Pages Service**.**update yp service (…)

The main logic is an iteration for the provision of a yellow pages service:

— Travel Terminal**.** yp request (…)

— Travel Terminal**.**read payment card/tag ([2214])

— Yellow Pages Service**.**yp transaction (…)

— Operating Interface**.**yp service order(…)

— Billing Record**.**create ([2214])

followed by use of the Payment Transaction (Figure 59 at line 3) and then

— Travel Terminal**.**confirm yp request (…)

Independent transactions arise in the case of a Traveller revising plans for the remaining trips of a journey. These are described in Pre-Journey Information (Figure 20 at line 3).

Independent transactions support a Driver on route. These are described in Route Guidance and Navigation (Figure 21 at line 23).

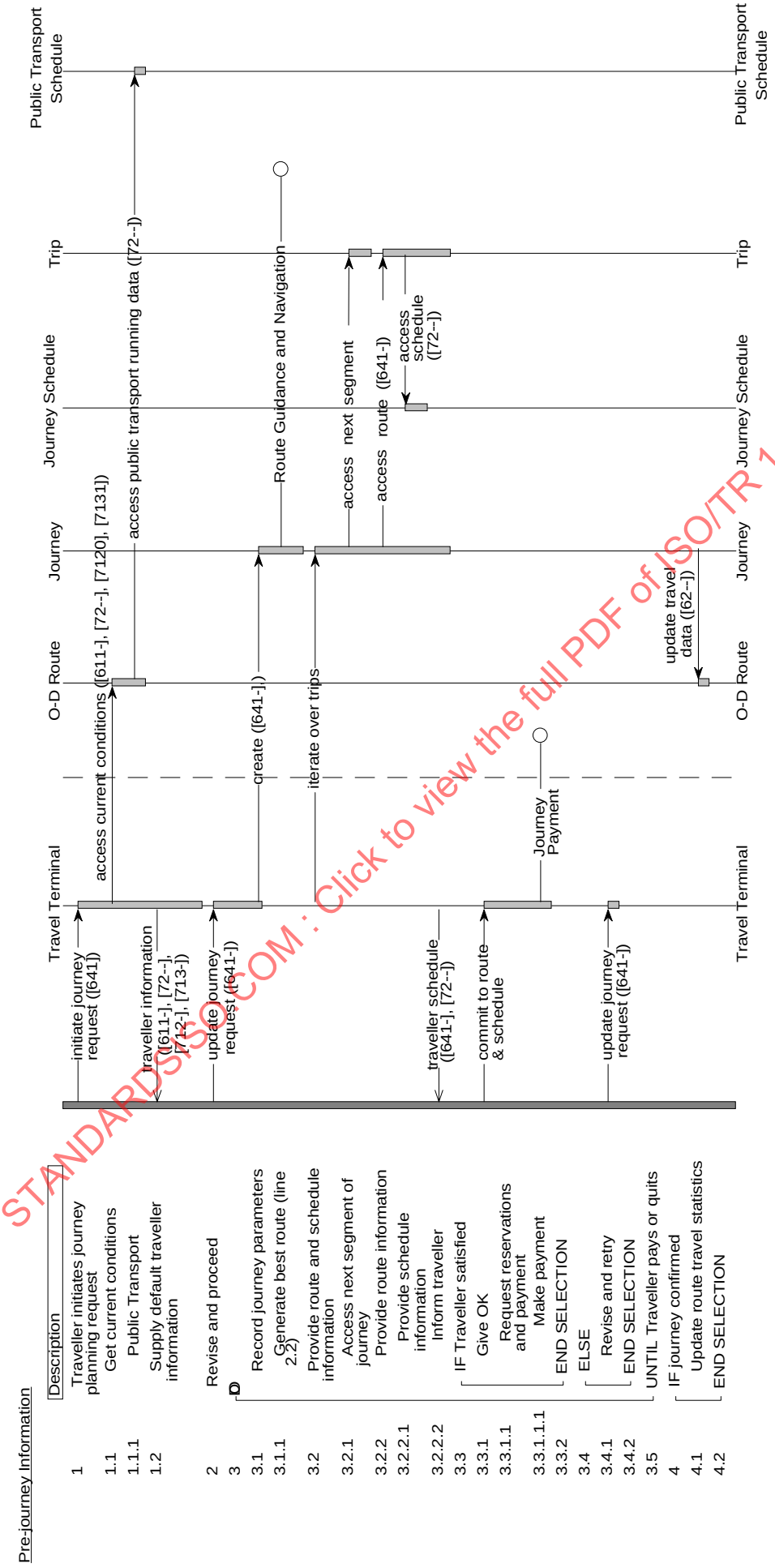

Figure 20 — Sequence diagram for Pre-journey Information

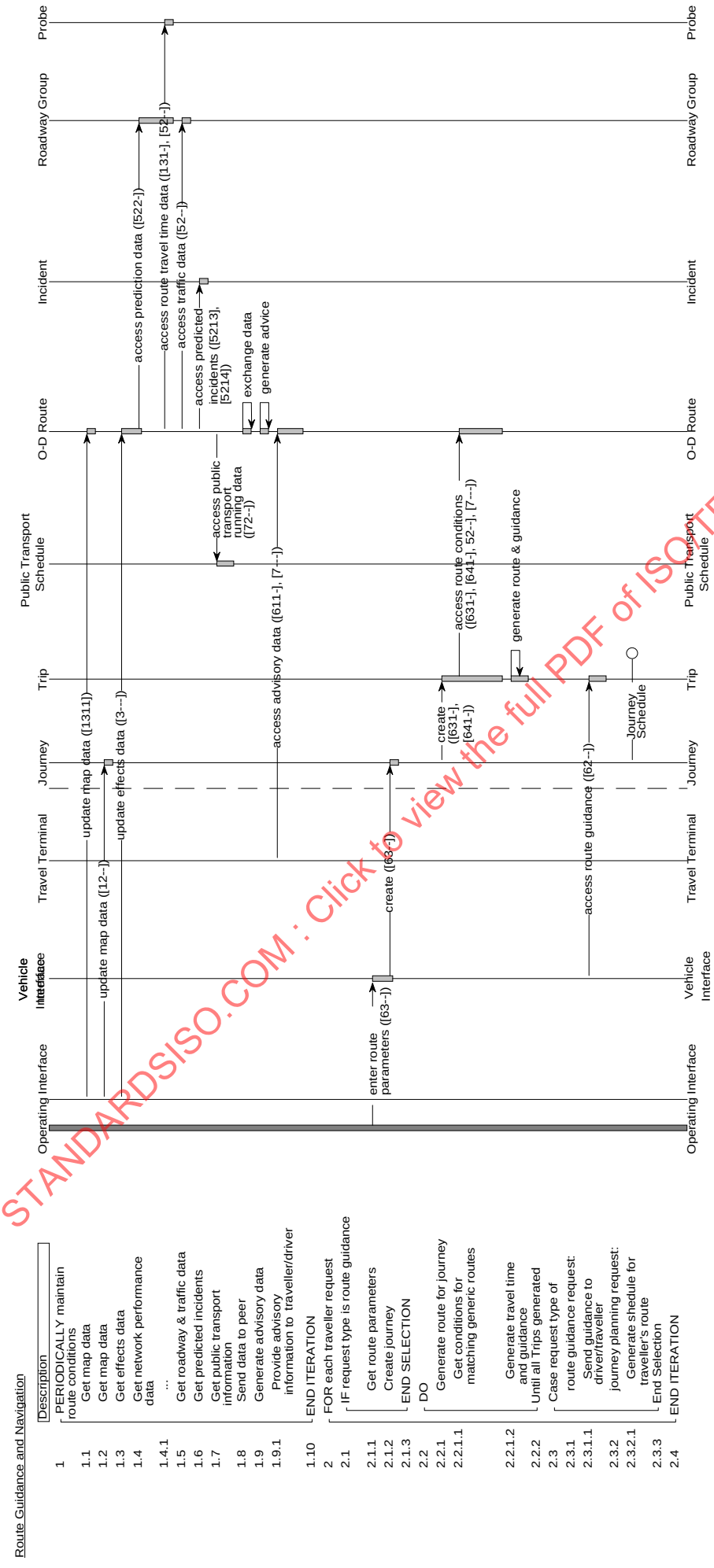

Figure 21 — Sequence diagram for Route Guidance and Navigation

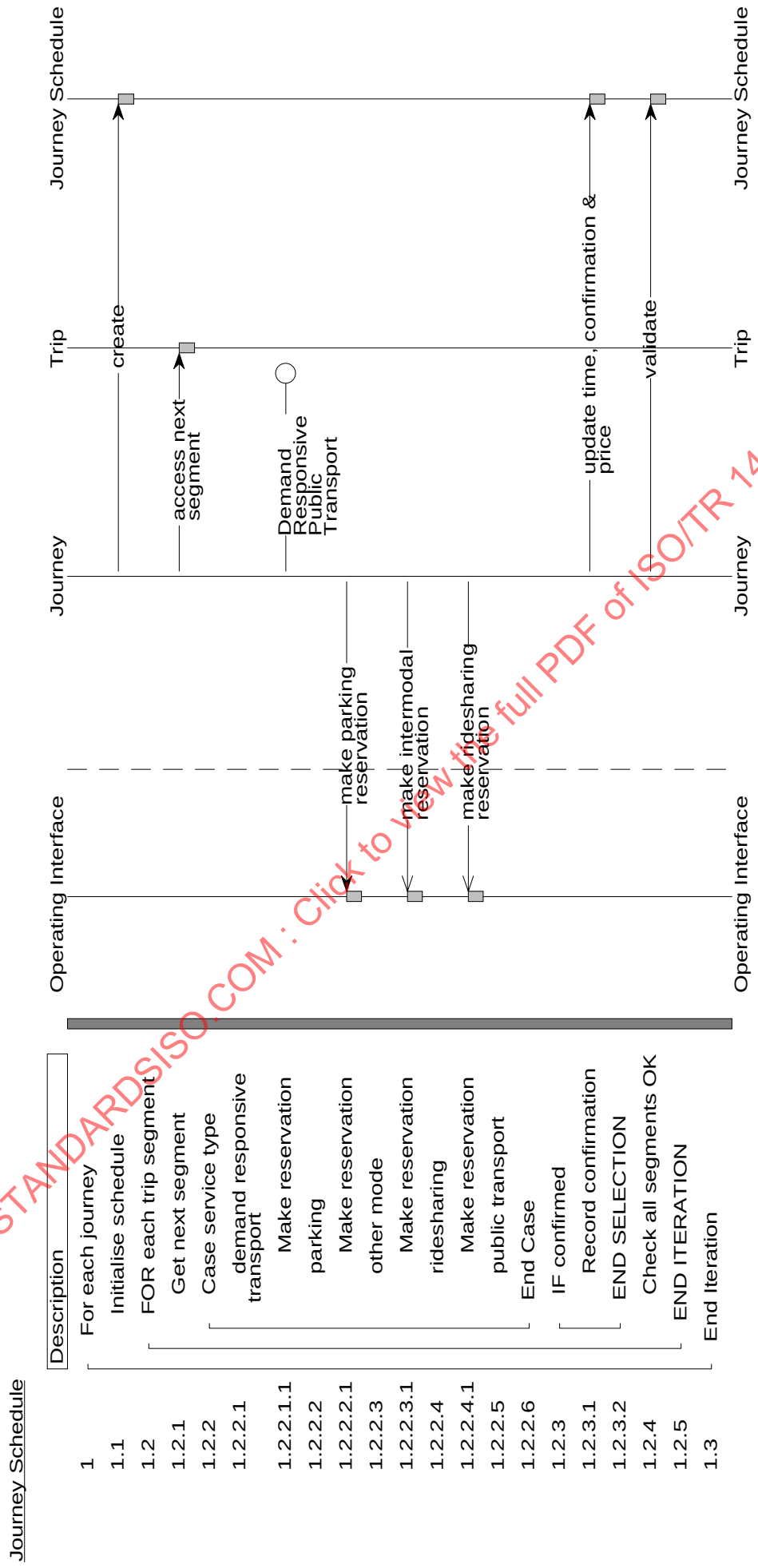

Figure 22 — Sequence diagram for Journey Schedule

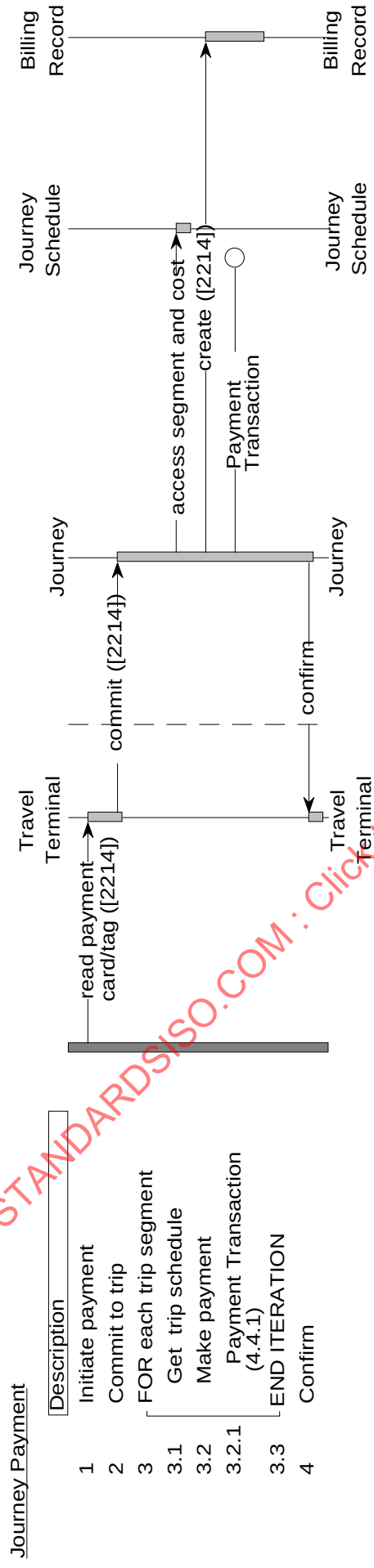

Figure 23 — Sequence diagram for Journey Payment

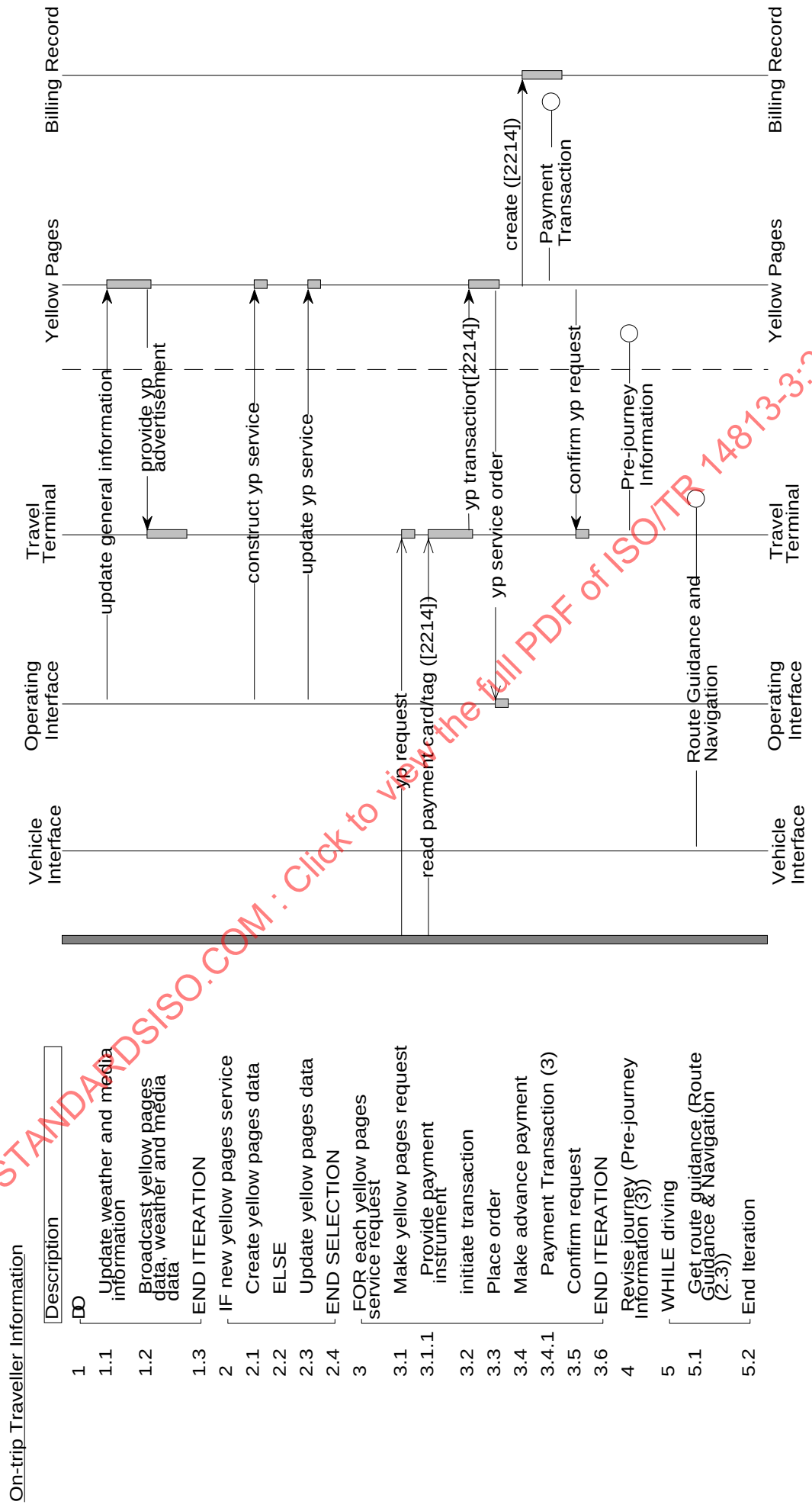

Figure 24 — Sequence diagram for On-trip Traveller Information

7.2 Traffic Management

The information classes identified in 6.3 were developed to meet the message and information contexts of traffic management and information centres. Thus they are highly relevant to this clause. Therefore the elaboration in this clause produces pairs of diagrams for each use case (e.g. Figures 25 and 26). There is one sequence diagram and one class diagram, the latter showing the key classes and parameter information classes which occur in the former diagram. The class diagram partitions the classes by package. In order to avoid cluttering the class diagrams it is usual to associate a parameter information class with a key class in only one package. There is always a key class with which each information class (parameter) can be associated.

7.2.1 Traffic and Pollution Measurement and Control

The abstract operations for Traffic and Pollution Measurement and Control in ISO/TR 14813 Part 2 are provided by the Roadway Class and the Event Class. This leads to the following key classes defined in 6.1.

— Local Control Group

— Roadway Group

— Violation

The following new information classes are required:

— Movement: This information class is invented to describe a stream of traffic and to record measurements about the associated traffic flow. The stream measured might be over a particular manoeuvre, or over a more loosely specified route beginning in a specified road element and ending in a specified road element. It is associated with the Local Control Group class.

— Parking: This information class is invented to record measurements about parking operations provided by Parking actors. It is associated with the Roadway Group class.

— Pollution: This information class is invented to record and process measurements about pollution. It is associated with the Incident class.

The following new interface classes are required:

— Roadside Peripheral: This interface class is invented to interface various signs, detectors and transceivers to TICS.

— Roadside Sensor: This interface class is invented to interface various detectors and transceivers to TICS. It is a subclass of Roadside Peripheral.

— Vehicle Sensor: This interface class is invented to interface special Automatic Vehicle Identification equipment to TICS. It is a subclass of Roadside Sensor.

The sequence diagram is shown in Figure 25. The main logic is a continuing iteration whereby Traffic is measured at the primitive level by sensors and then these measurements are integrated and aggregated in stages by objects of the Local Control Group class and the Roadway Group class to develop the strategic traffic measurements. In parallel with these transactions any violation of regulations by individual vehicles is detected and acted upon.

There are some less frequent maintenance transactions performed, dealing with reference geographic data, operator data display and processing violations. These are placed outside the main loop.

Comments on some of the interactions of the sequence diagram in Figure 25 are given below.

Objects of the interface classes Roadside Sensor and Vehicle Sensor interface between sensor hardware and an object of the Local Control Group class. They invoke two operations in the Local Control Group class:

— Local Control Group.apply traffic regulations ([411-], [412-], [414-])

— Local Control Group.update sensor measurements ([411-], [412-])

In addition an operation in the Pollution class is invoked to derive the area pollution:

— Pollution.update pollution data ([414-])

Annex A contains documentation on each of the parameters, which can also be identified in Figures 11 to 19.
If the operation, Local Control Group.apply traffic regulations, determines that an infringement has occurred then the following operations are invoked:

— Violation.create ([411-], [412-], [414-])

— Vehicle Interface.warn driver ([6132])

Objects of the Local Control Group class invoke operations in an object of the Roadway Group to process measurements at the higher level:

— Roadway Group.integrate measurements ([41--])

— Roadway Group.update current data ([412-], [413-])

— Movement.update measurements ([412-])

The Pollution class updates its higher level measurements:

— Pollution.update current data ([414-])

At the end of the iteration any adjustment to the attributes of objects of the Local Control Group or Sensor class is requested:

— Local Control Group.update control data ([51--])

— Roadside Sensor.update control data (…)

— Roadside Peripheral.update control data (…)

The above operations are just two of numerous examples of missing parameters, with no requisite data elements in draft ISO 14827.

Traffic operators may select and display data:

— Roadway Group.support operator access ([4---])

The final stage is used to process any violation and notify the conformance agency:

— Violation.process violation ([81--])

— Operating Interface.report violation ([81--])

The elaboration of the classes of the Roadway and Events packages developed in the above process is summarised in Figure 26, with the key classes denoted by double rectangles. Associations already shown in diagrams in clause 5 may not be included.

7.2.2 Performance Evaluation

The abstract operations for Performance Evaluation in ISO/TR 14813 Part 2 are provided by the Roadway Class, the Transport Class and the Event Class. This leads to the following key classes defined in previous clauses.

— Roadway Group

— Parking

— O-D Route

— Incident

— Pollution

— Probe

The sequence of object interactions for the Performance Evaluation use case is shown in Figure 27. The main part is a continuing low frequency iteration whereby a wide range of performance indicators are gathered and assessed. Actions to modify the active control strategies may be taken following this analysis, and the derived performance data is stored for long term reference.

A maintenance transaction keeps the reference geographic data current.

The first part of the iteration is to gather all the relevant current data on which network performance will be assessed. Thus the transactions are dependent on those of Traffic and Pollution Measurement and Control. The relevant data associated with other key classes is retrieved by the operations:

— Incident.access incident data ([42--])

— Probe.access route travel time data ([5211])

— Pollution.access pollution data ([414-])

— O-D Route.access travel data ([63--], ([64--])

— O-D Route.access route data ([412-])

— Parking.access parking data ([713-])

The key computational operation is “analyse data” of the Roadway Group class. Note this analysis would be carried out at the individual object level (corresponding to a region analysis) and at the static (class) level (corresponding to a global analysis across regions).

The remaining operations of the loop are responsible for storing the performance statistical data for this period for long term reference:

— Roadway Group.update performance data ([94--])

— Roadway Group.update traffic control data ([51--])

— Roadway Group.update long term data ([4-], [5---], [94--])

— Incident.update long term data ([3---])

— Pollution.update long term data ([414-])

— O-D Route.update long term data ([521-])

— Parking.update long term data ([713-])

The elaboration of the classes of the architecture is summarised in Figure 28, with the key classes denoted by double rectangles.

7.2.3 Performance Prediction

The abstract operations for Performance Evaluation in ISO/TR 14813 Part 2 are provided by the Roadway Class, the Transport Class and the Event Class. This leads to the following key classes defined in previous clauses.

— Roadway Group

— O-D Route

— Incident

The sequence of object interactions for the use case is shown in Figure 29. The iteration uses the Performance Evaluation transactions.

Preceding the main computation the incident data is retrieved and the performance data exchanged between Roadway Group objects:

— Incident.access predicted incidents ([5213], [5214])

— RoadwayGroup.exchange data ([51--], [521-])

The key computational operation is:

— Roadway Group.predict network performance ([522-])

Note this analysis would be carried out at the individual object level (corresponding to a region analysis) and at the static (class) level (corresponding to a global analysis across regions).

The remaining operations of the loop are responsible for distributing the prediction data:

— O-D Route.update prediction data ([522-])

— Roadway Group.exchange predictions ([522-])

The elaboration of the classes of the architecture in the above process is summarised in Figure 30, with the key classes denoted by double rectangles.

7.2.4 Traffic Control

The abstract operations for Traffic Control in ISO/TR 14813 Part 2 are provided by the Roadway Class. This leads to the following key classes defined in previous clauses.

— Roadway Group

— Local Control Group

— Incident

— Parking

The following new information class is required:

— Control Plan: This information class is invented to maintain the traffic control strategy and operational parameters which can be implemented over the roadway network contained by one or more Roadway Groups. It is associated with the Roadway Group control class.

The following new interface class is required:

— Special Vehicle Interface: This information class is invented to interface special vehicle drivers to TICS.

The sequence of object interactions for the Traffic Control use case is shown in Figure 31. Information is provided to Operators.

— Operating Interface.operator data output ([41--])

Traffic Operators can override control assynchronously:

— Roadway Group.update control strategy ([51--])

The main part of the sequence is a continuing low frequency iteration with an inner higher frequency iteration. The sequence depends on the transactions of Performance Evaluation and Performance Prediction. The low frequency transactions then support changes to the current control strategy:

— Control Plan.compute plan ([3---], [4---], [51--])

— Roadway Group.update incident strategy ([51--])

— Local Control Group.update control data ([51--])

— Parking.update control data (...)

— Operating Interface.operator data output (...)

The control loop may invoke the transactions of Infrastructure Maintenance Management to perform infrastructure changes.

Conformant or autonomous control transactions use the operation:

— Local Control Group. generate control parameters for peripherals (...)

High frequency control transactions involving the Roadside Peripherals are based on the operations:

— Local Control Group.request priority ([211-])

— Local Control Group.update rail traffic data (…)

— Roadside Peripheral.update control data (...)

— Operating Interface.notify rail (…)

The elaboration of the classes of the architecture in the above process is summarised in Figure 32, with the key classes denoted by double rectangles.

7.2.5 Incident Management

The abstract operations for Incident Management in ISO/TR 14813 Part 2 are provided by the Roadway Class and the Event Class. This leads to the following key classes defined in previous clauses.

— Roadway Group

— Incident

— Pollution

— Emergency

The following new control class is required

— Commercial Vehicle Tour: This control class is invented to maintain and execute the plan for a commercial vehicle tour.

The following new information class is required

— Incident Response: This information class is invented to maintain response plans to be used in incident management.

The sequence of object interactions for the Incident Management use case is shown in Figure 33. Operators asynchronously develop the Incident Response plans:

— Incident Response.create ([531-], [9614])

The next part is a regular iteration beginning with the transactions of Traffic and Pollution Measurement and Control. Then the other sources of information provide additional data using the generic operation:

— Roadway Group.update incident data ([2240], [414-], [44--], [3---], [532-], [542-])

Once all incident data has been gathered it is analysed:

— Roadway Group.data fusion (...)

If this analysis and reasoning requires action, the following transactions apply:

— Incident.create ([2240], [3---], [44--], [532-], [54--])

— Incident Response.match incident ([2240], [3---], [44--], [53--])

— Incident.update response data ([531-])

Then the Traffic Control cycle is exercised to put the response into effect. If other responses are required an Emergency object is created:

— Emergency.create([44--])

Finally any traffic control recommendations are sent to strategic peers:

— Roadway Group.exchange data ([51--])

The elaboration of the classes of the architecture in the above process is summarised in Figure 34, with the key classes denoted by double rectangles.

7.2.6 Demand Management

The abstract operations for Demand Management in ISO/TR 14813 Part 2 are provided by the Roadway Class, the Transport Class and the Payment Class. This leads to the following key classes defined in previous clauses.

— Roadway Group

— O-D Route

— Parking

— Public Transport Schedule

— Tariff

The following new information class is required

— Demand Plan: This information class is invented to maintain the demand management strategy and parameters which can be implemented over the roadway network contained by one or more Roadway Groups. It is associated with the Roadway Group control class.

The sequence of object interactions for the Demand Management use case is shown in Figure 35. The operator maintains the demand response component of control plans by asynchronous interactions.

— Demand Plan.update demand policy data(…)

The main management transactions are executed periodically. This is based on the Performance Prediction transactions and recent traffic messages:

— O-D Route.access travel data ([61--])

— Roadway Group.predict demand (…)

This is followed by transactions which determine the action to be taken including Operator overrides.

— Demand Plan.compute plan([3---], [4---], [52--], [61--], [7---])

— Roadway Group.update demand strategy(…)

Transactions which might be executed subsequently are:

— Parking.request to cater to demand(...)

— Operating Interface.operator data output (...)

— Tarrif.request to cater to demand(...)

— Operating Interface.operator data output (...)

— Public Transport Schedule.request service change(...)

— Operating Interface.operator data output (...)

The parameters for the operations are not addressed in ISO 14827.

The elaboration of the classes of the architecture in the above process is summarised in Figure 36, with the key classes denoted by double rectangles.

7.2.7 Transport Planning Support

The abstract operations for Transportation Planning Support in ISO/TR 14813 Part 2 are provided by the Roadway Class, the Transport Class, and the Event Class. This leads to the following key classes defined in previous clauses.

— Roadway Group

— Parking

— Control Plan

— Probe

— O-D Route

— Public Transport Schedule

— Commercial Vehicle Tour

— Incident Response

— Pollution

The following new information classes are required

— Emergency Vehicle: This class is invented to maintain information about those vehicles used in emergency operations

— Commercial Vehicle: This class is invented to maintain information about those vehicles used in commercial operations

The sequence of object interactions for the Transport Planning Support use case is shown in Figure 37. A planning session is typically based on Performance Evaluation. Then current and historical data relevant to the particular planning task is accessed:

— Emergency Vehicle.access fleet operational data (...)

— Commercial Vehicle Tour.access fleet operational data ([73--])

— Probe.access route travel time data ([52--])

— Parking.access parking data ([713-])

— Pollution.access pollution data ([414-])

— Public Transport Schedule.access public transport operational data ([222-], [72--])

— O-D Route.access travel data ([62--])

— O-D Route.access route data ([41--])

The key computational operation is “decision support” of the Roadway Group class. Note this analysis would be carried out at the individual object level (corresponding to a region analysis) and at the static (class) level (corresponding to a global analysis across regions).

The remaining operations of the loop are responsible for making any new strategy:

— Control Plan.create ([51--])

— Demand Plan.create (…)

— Incident Response.create ([531-])

The elaboration of the classes of the architecture in the above process is summarised in Figure 38, with the key classes denoted by double rectangles.

7.2.8 Infrastructure Maintenance and Management

The abstract operations for Infrastructure Maintenance and Management in ISO/TR 14813 Part 2 are provided by the Roadway Class and the Event Class. This leads to the following key classes defined in previous clauses.

— Roadway Group

— Incident

The sequence of object interactions for the Infrastructure Maintenance and Management use case is shown in Figure 39.

A transaction involves the development of the maintenance plan based on network geographic data held in map information classes associated with the Roadway Group. Future or immediate incidents are notified. Operators are instructed on actions required:

— Roadway Group.schedule maintenance ([3---])

— Incident.create ([3---])

— Operating Interface.instruct operator(…)

The elaboration of the classes of the architecture in the above process is summarised in Figure 40, with the key classes denoted by double rectangles.

7.2.9 Package Classes for Traffic Management

This clause summarises the architecture elaboration for Traffic Management at the package level. The class diagrams in this sub-clause bring together all the operations and information class associations identified in 7.2 for each package. The operations listed for each key class are required in one or more Traffic Management use case. Similarly the information classes occur in object interactions involving classes of the particular package. The generalised class is used to imply any sub-class of that class in order to avoid clutter.

Only three packages have a significant role in Traffic Management:

— Roadway (Figure 41)

— Events (Figure 42)

— Transport (Figure 43)

Note the Vehicles package is about vehicle operation and safety.

Again it should be noted that the package architectures in Figures 41 to 43 are incomplete for several reasons.

— The elaboration is based on only one of the eight use case diagrams

— Information classes (and parameters) are restricted to those defined in draft ISO 14827.

Furthermore there are information dependencies which span other use case. For example, the Probe key class has no operation to generate objects in the information class [52—] travelTime. The Vehicle Charges use cases would give rise to such operations. For this reason the Resource key class does not appear in Figure 41.

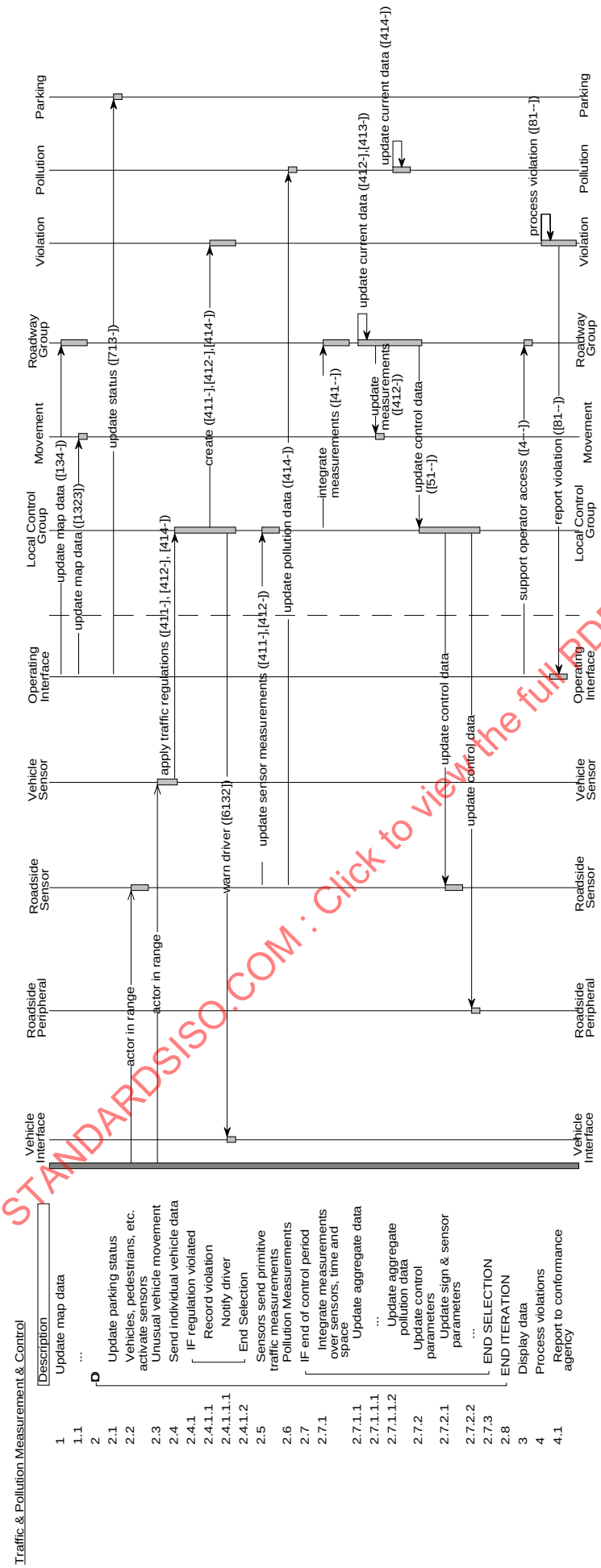

Figure 25 — Sequence diagram for Traffic and Pollution Measurement and Control

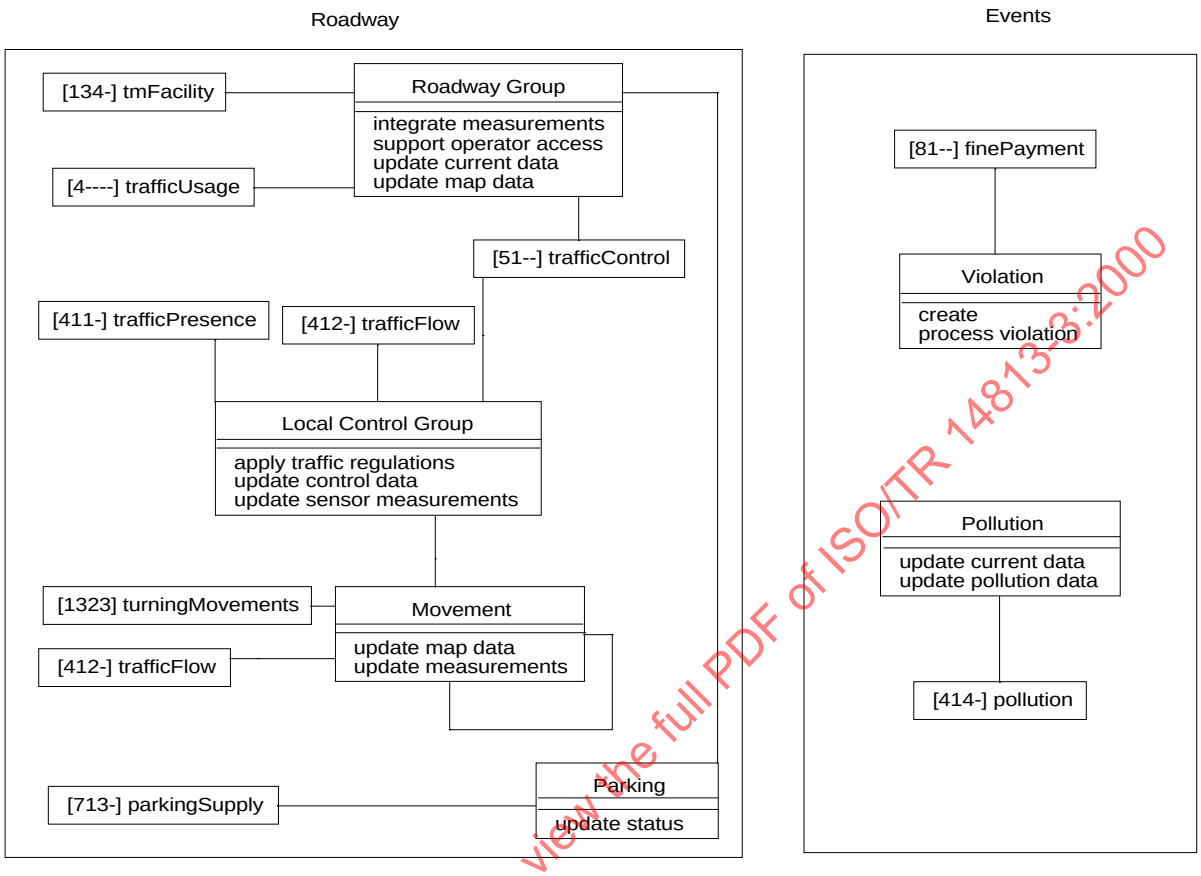

Figure 26 — Packages, control and information classes for Traffic and Pollution and Measurement and Control

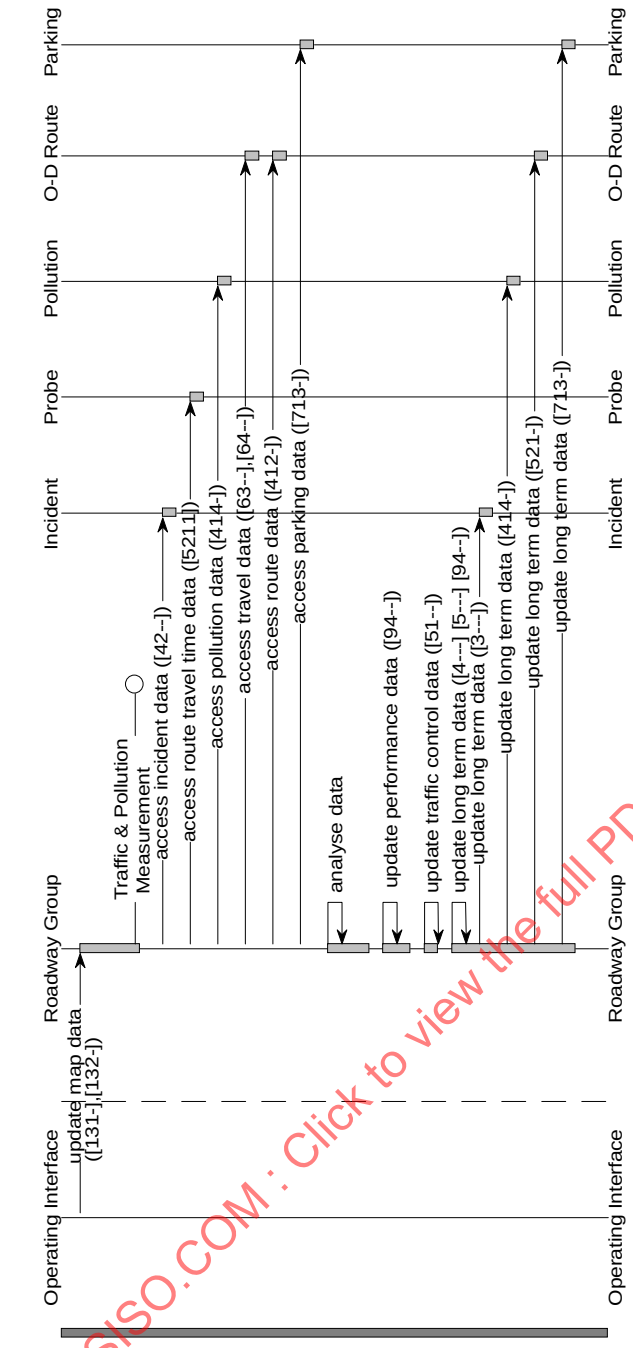

Figure 27 — Sequence diagram for Performance Evaluation

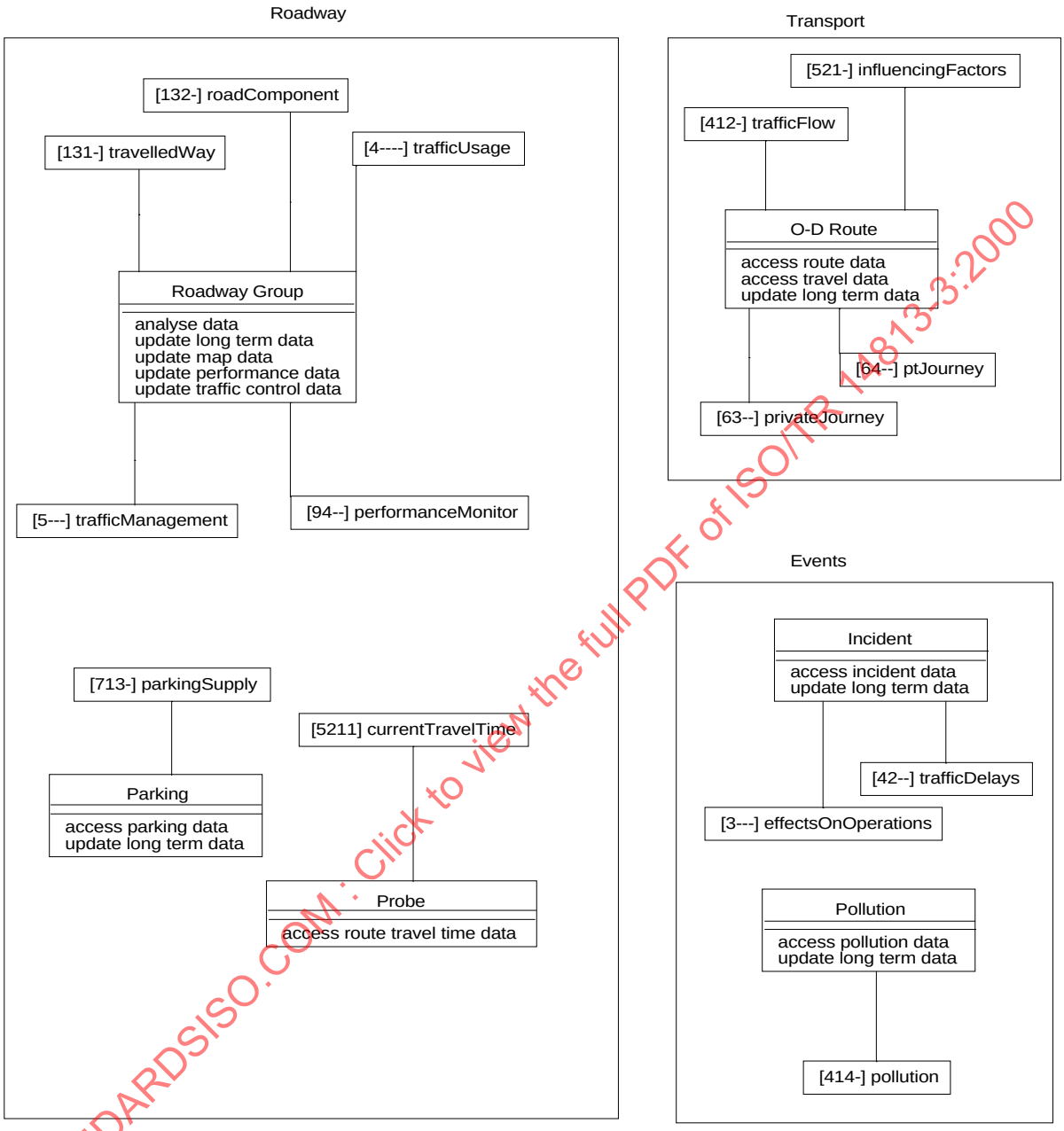

Figure 28 — Packages, control and information classes for Performance Evaluation

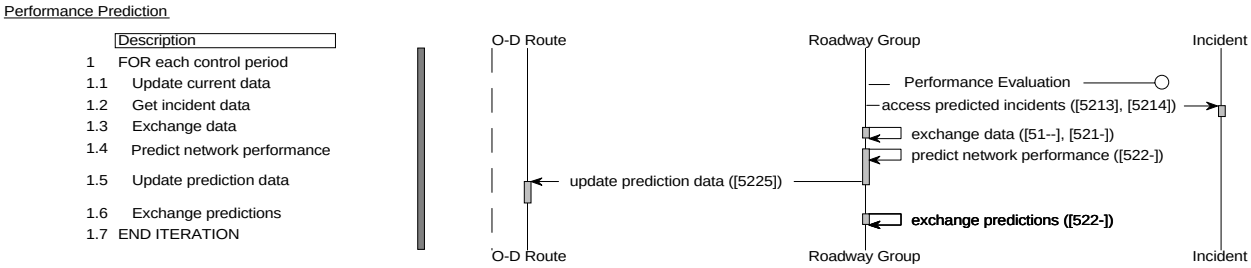

Figure 29 — Sequence diagram for Performance Prediction

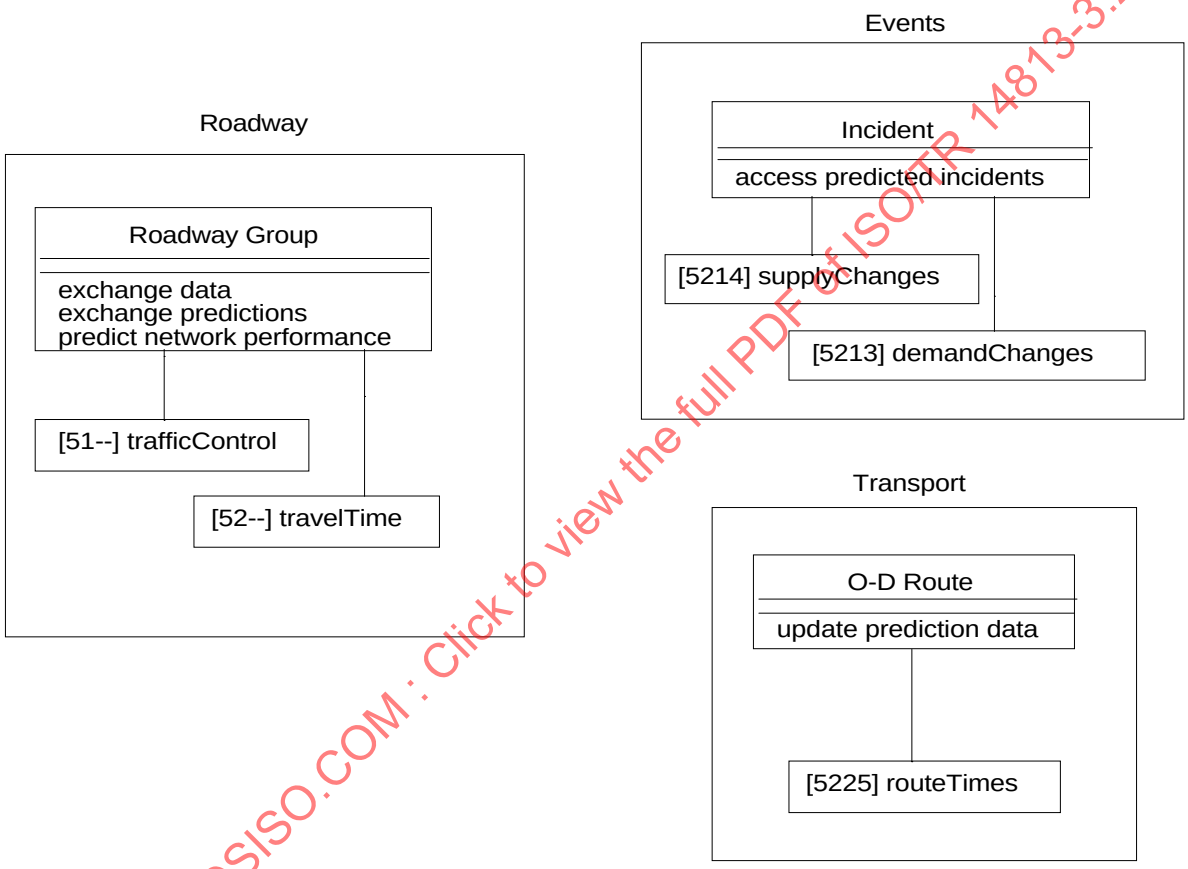

Figure 30 — Packages, control and information classes for Performance Prediction

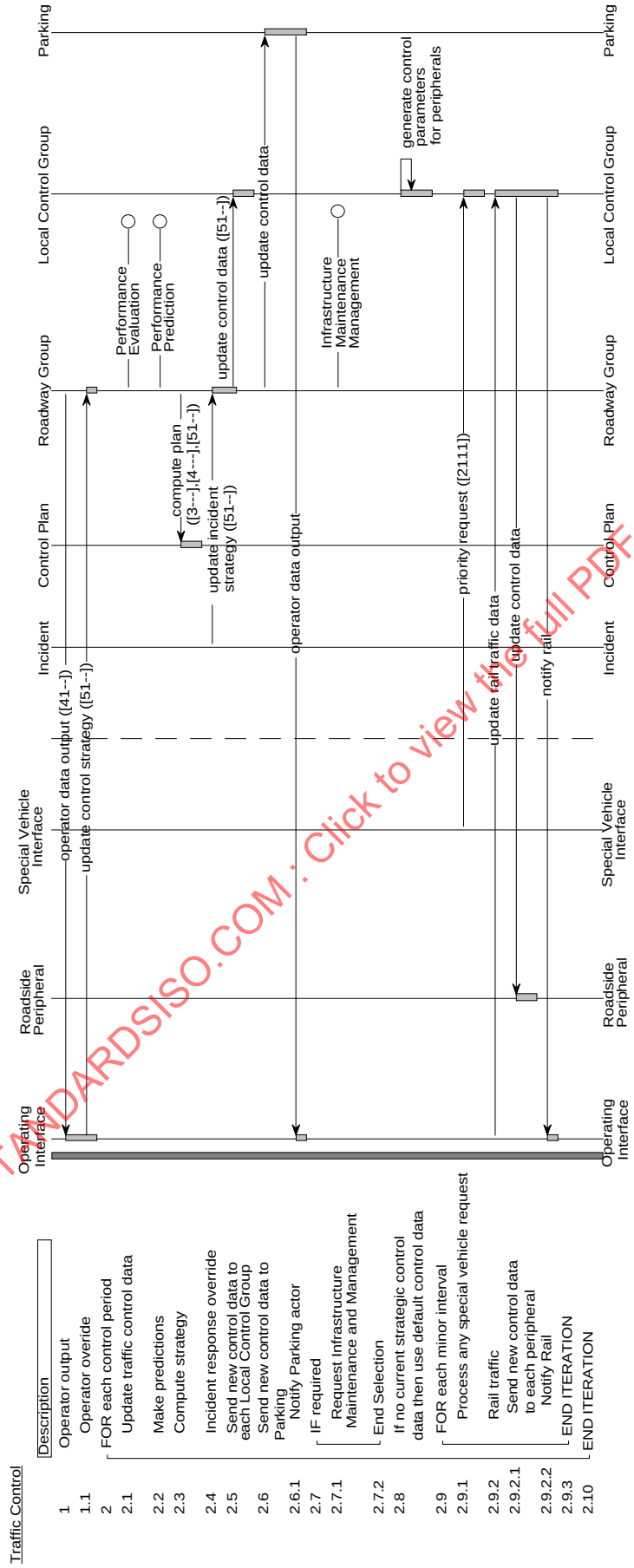

Figure 31 —Sequence diagram for Traffic Control

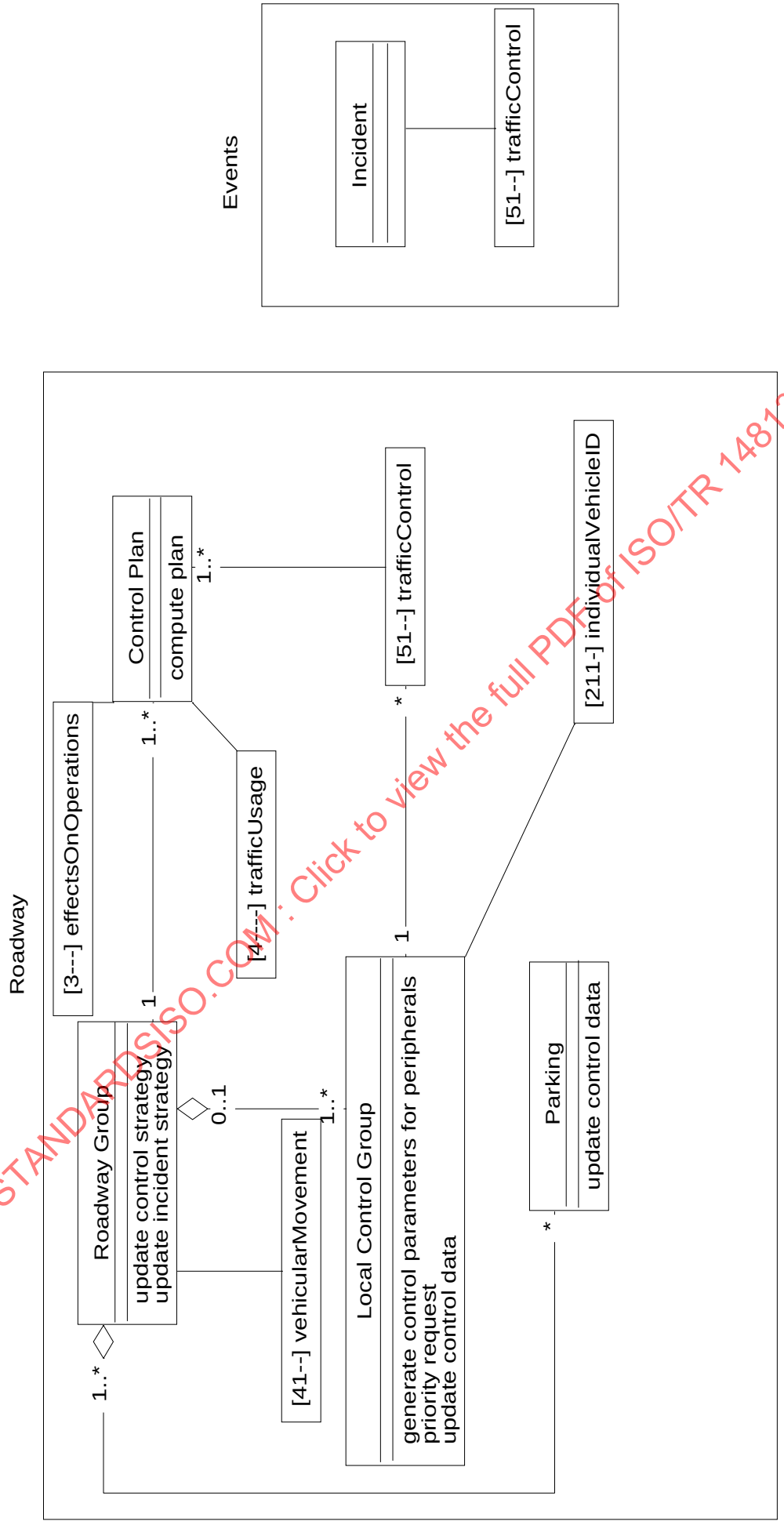

Figure 32 — Packages, control and information classes for Traffic Control

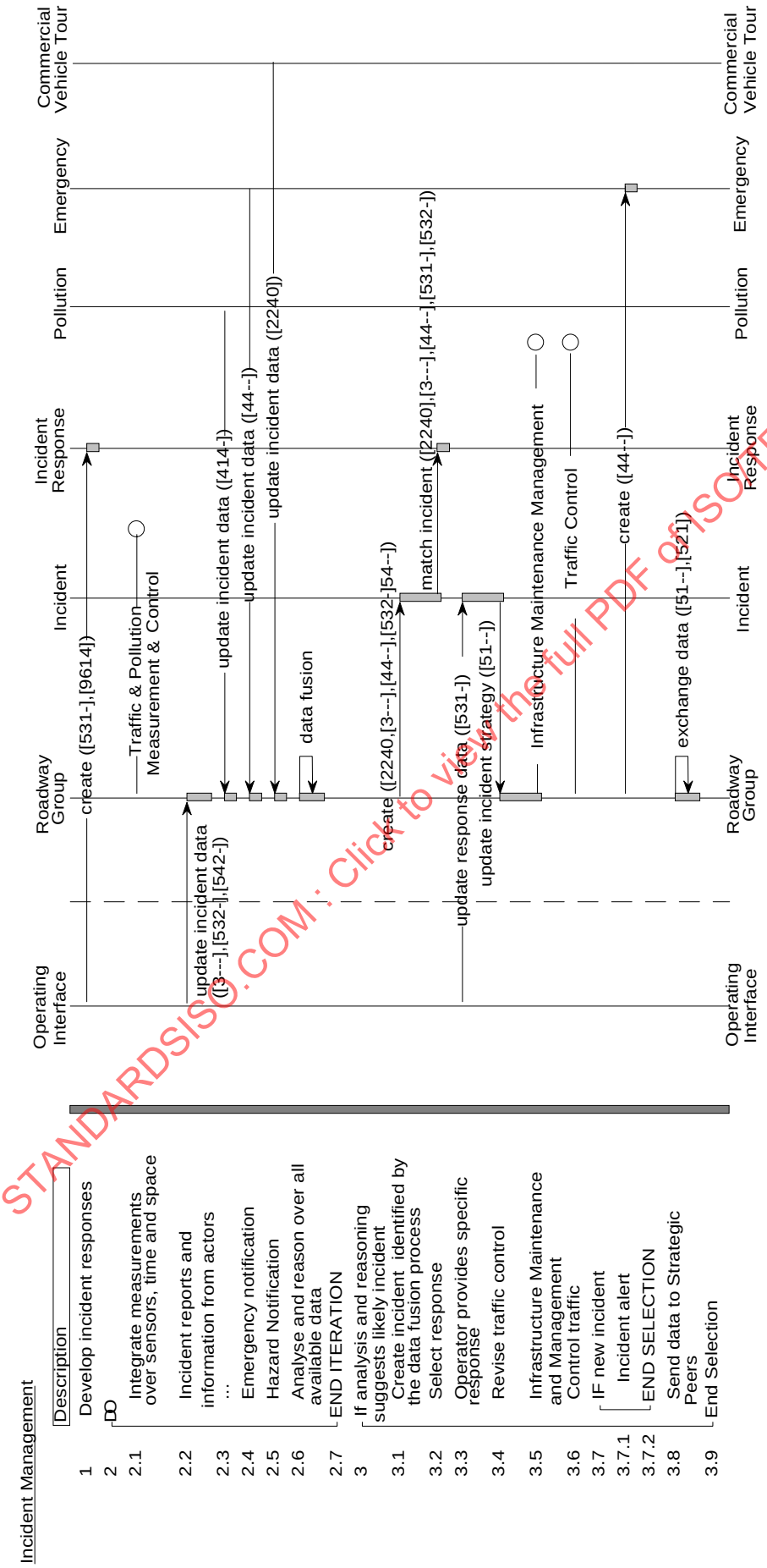

Figure 33 — Sequence diagram for Incident management

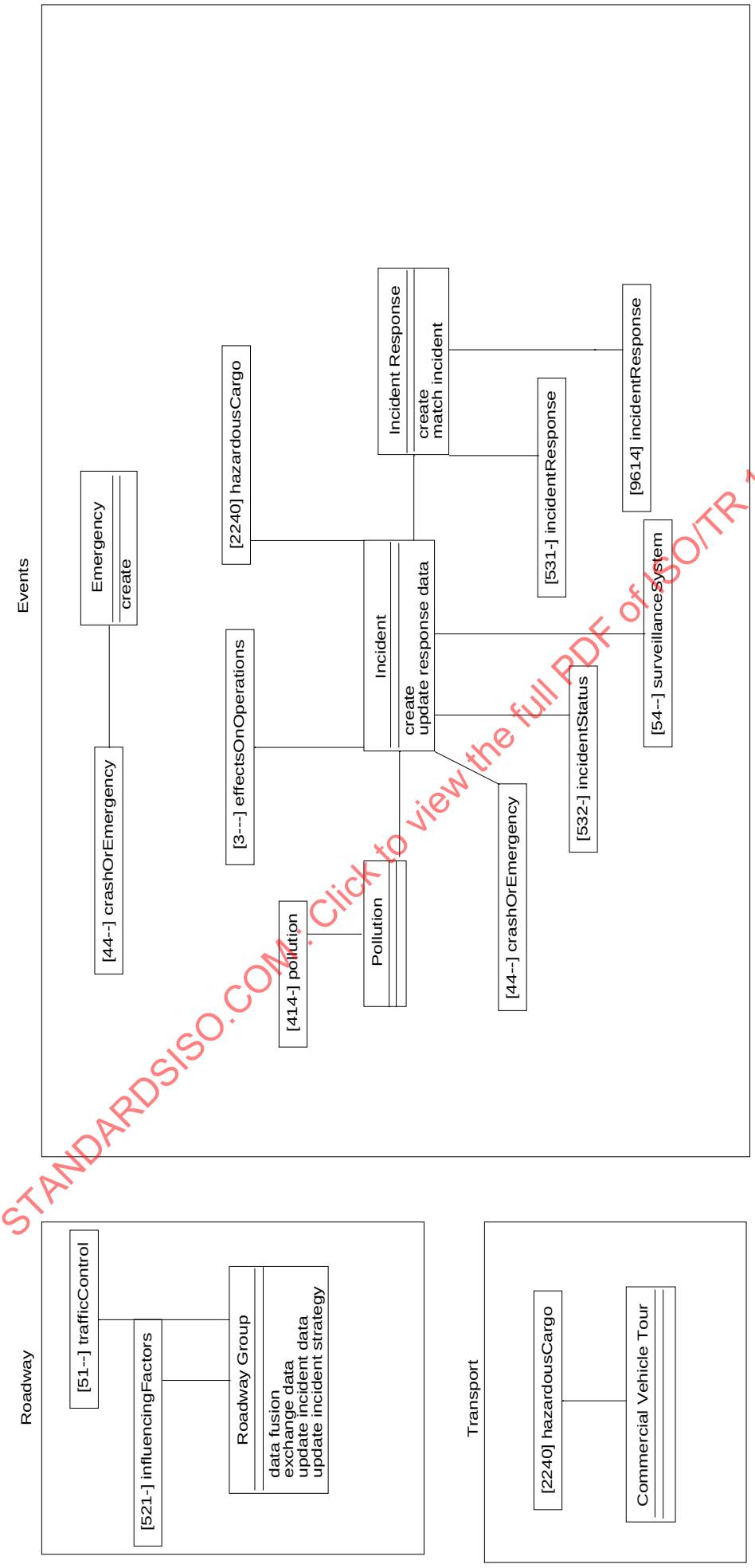

Figure 34 — Packages, control and information classes for Incident Management

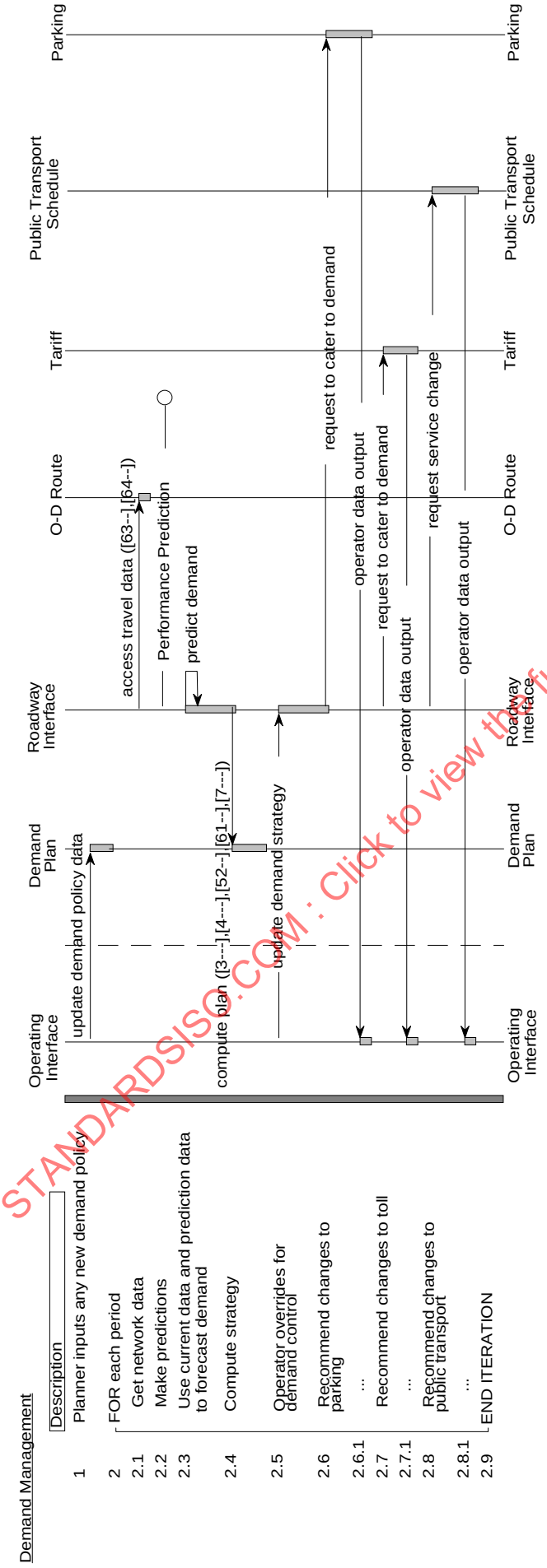

Figure 35 — Sequence diagram for Demand Management

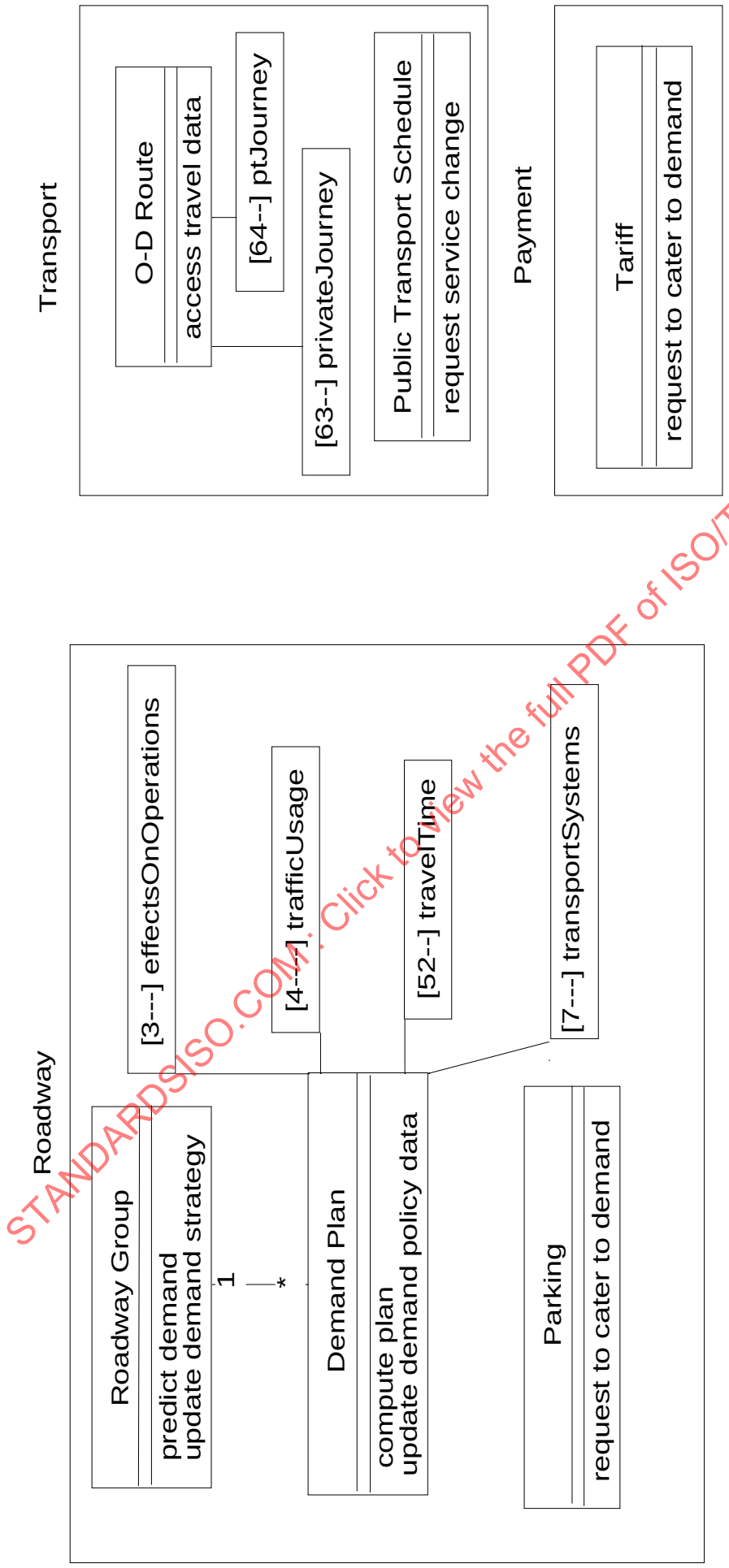

Figure 36 — Packages, control and information classes for Demand Management

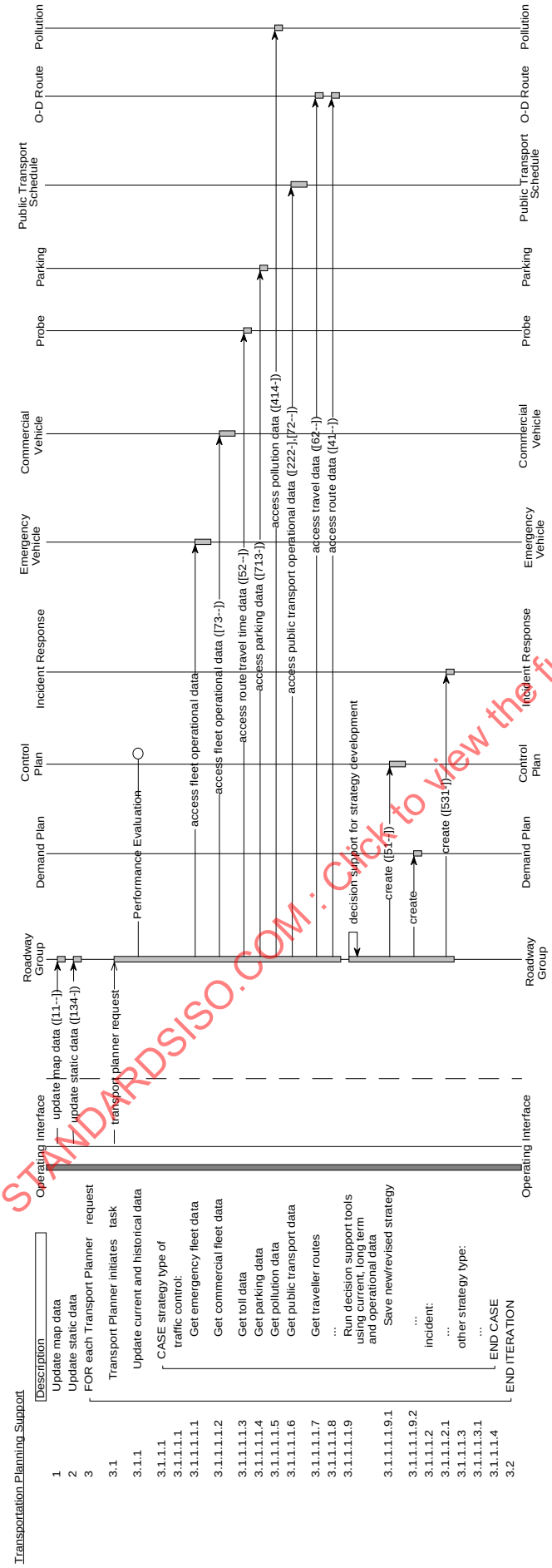

Figure 37 — Sequence diagram for Transportation Planning Support

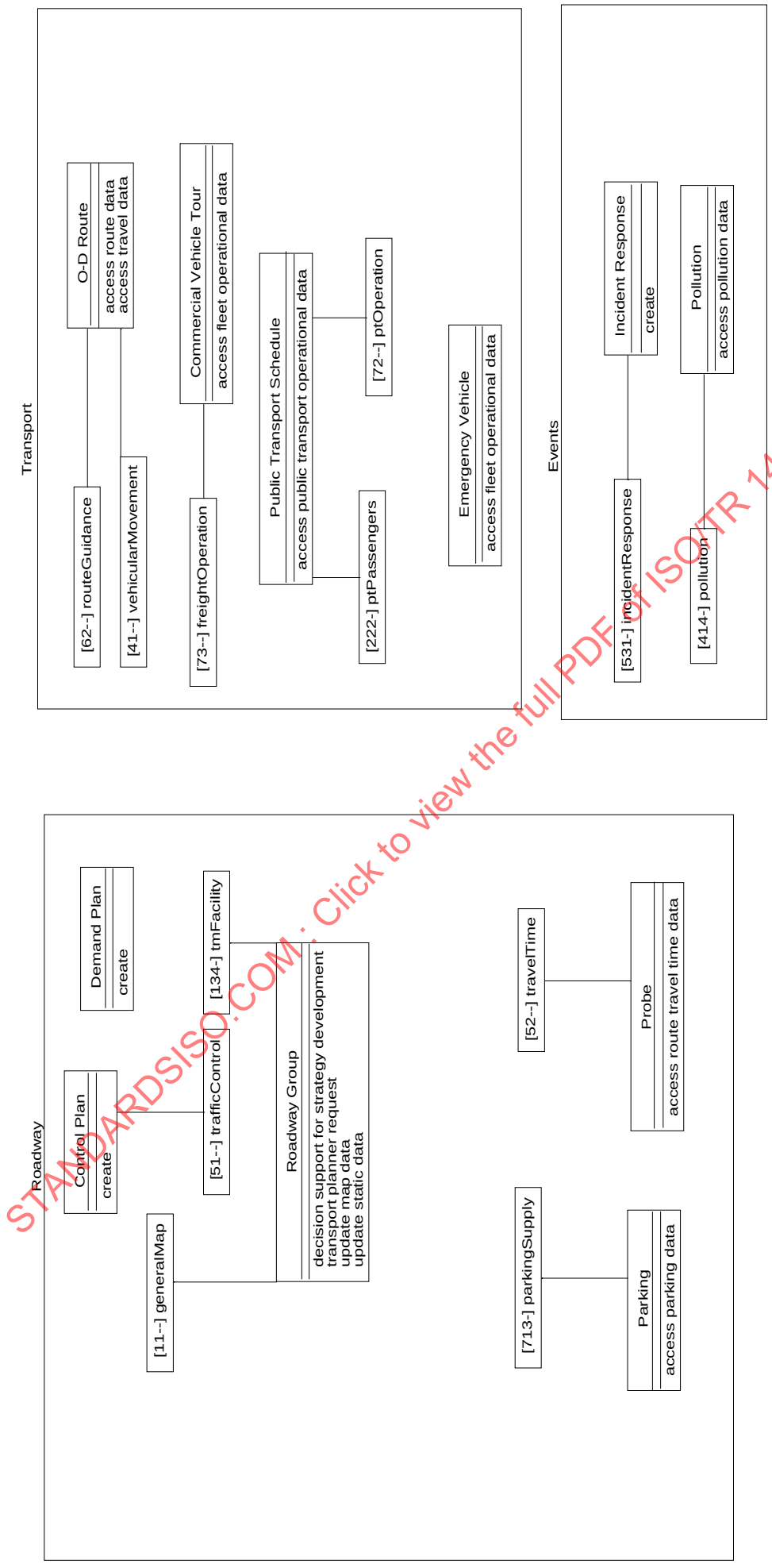

Figure 38 — Packages, control and information classes for Transportation Planning Support

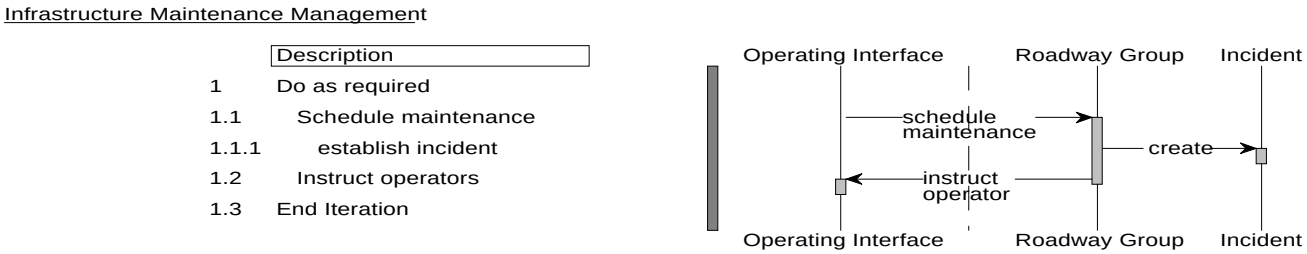

Figure 39 — Sequence diagram for Infrastructure Maintenance Management

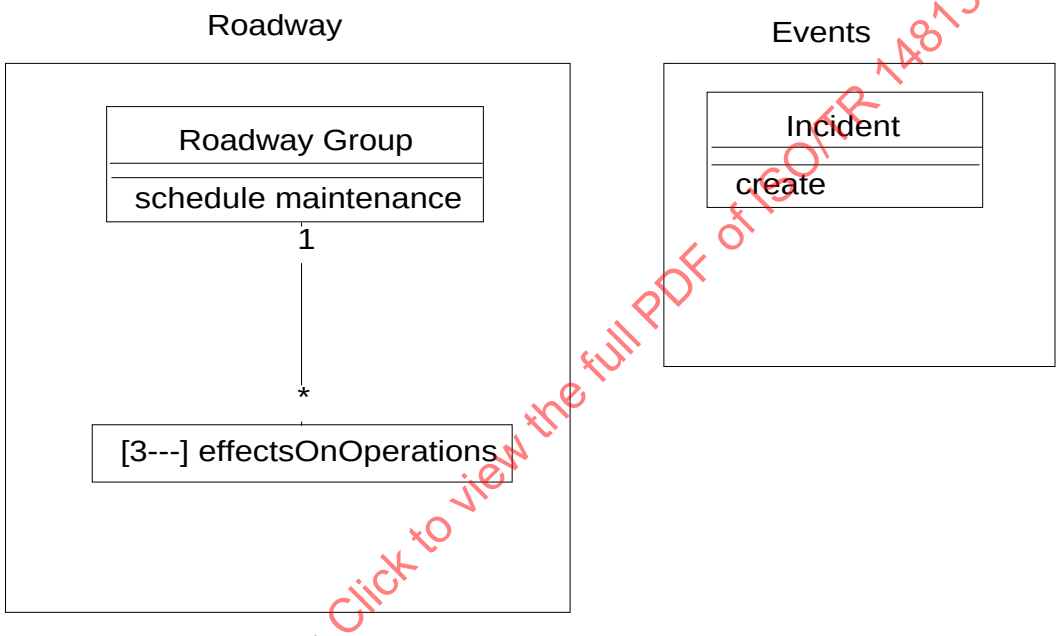

Figure 40 — Packages, control and information classes for Infrastructure Maintenance and Management

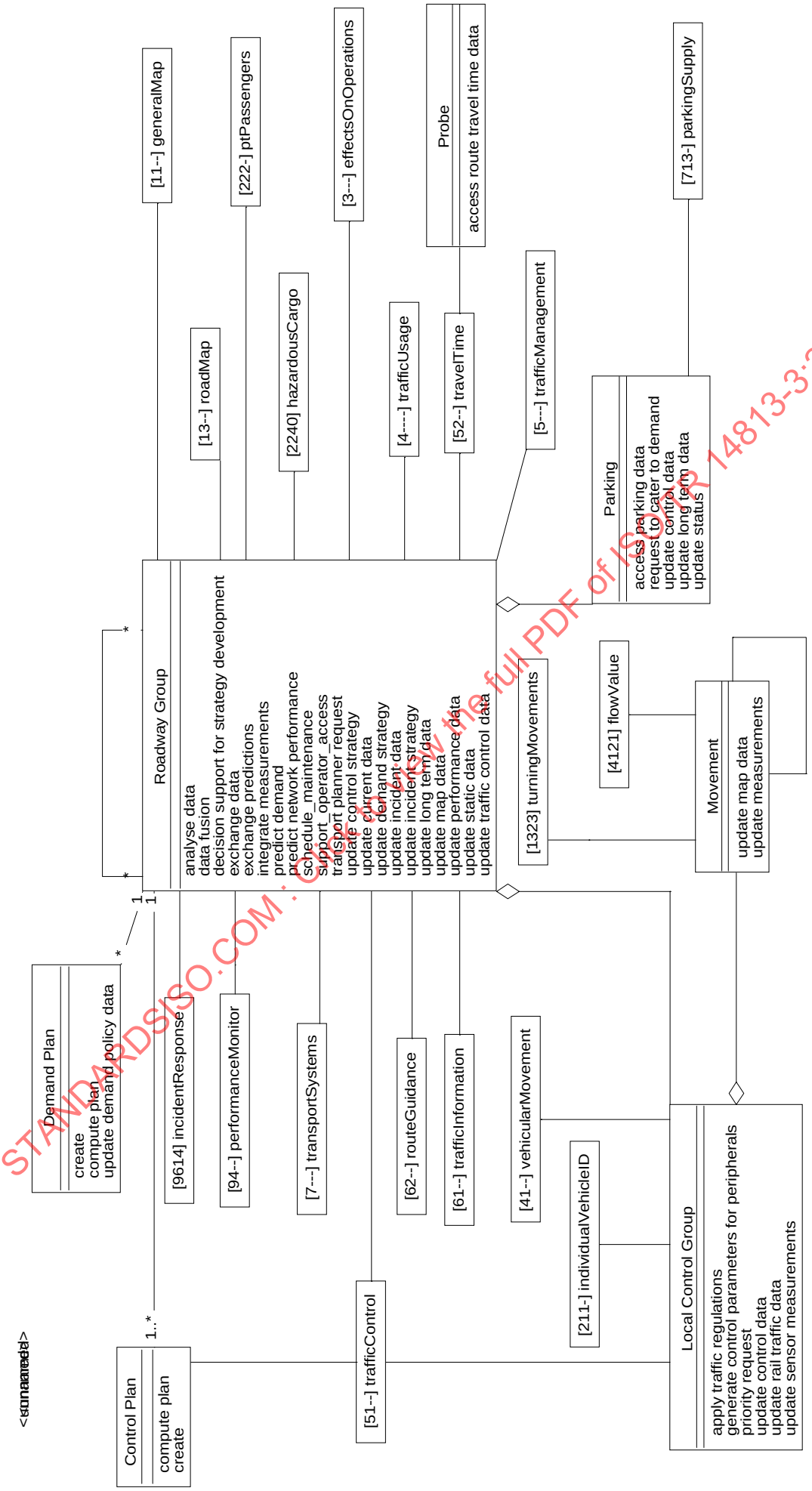

Figure 41 — Roadway package class operations and information class associations for Traffic Management

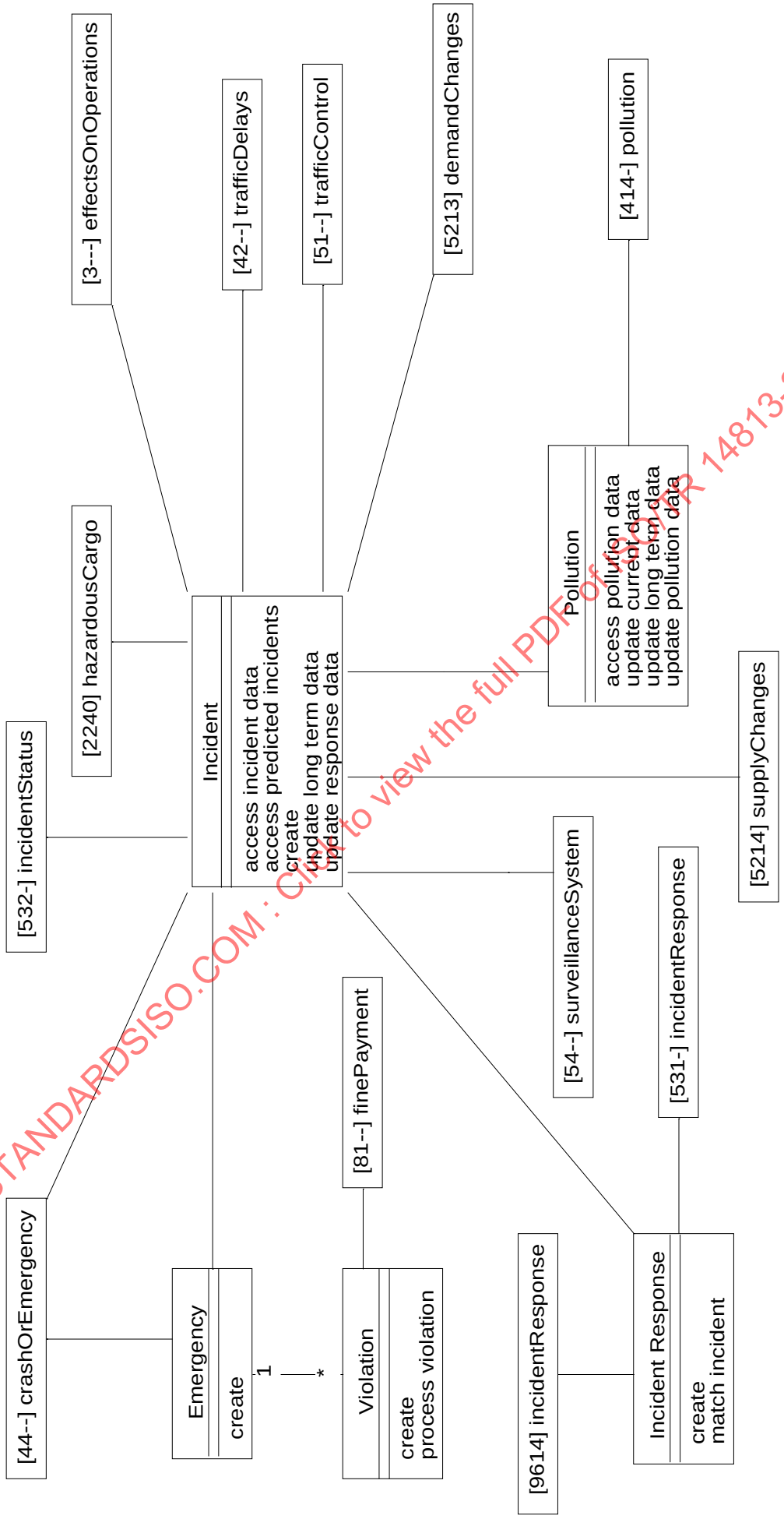

Figure 42 — Events package class operations and information class associations for Traffic Management

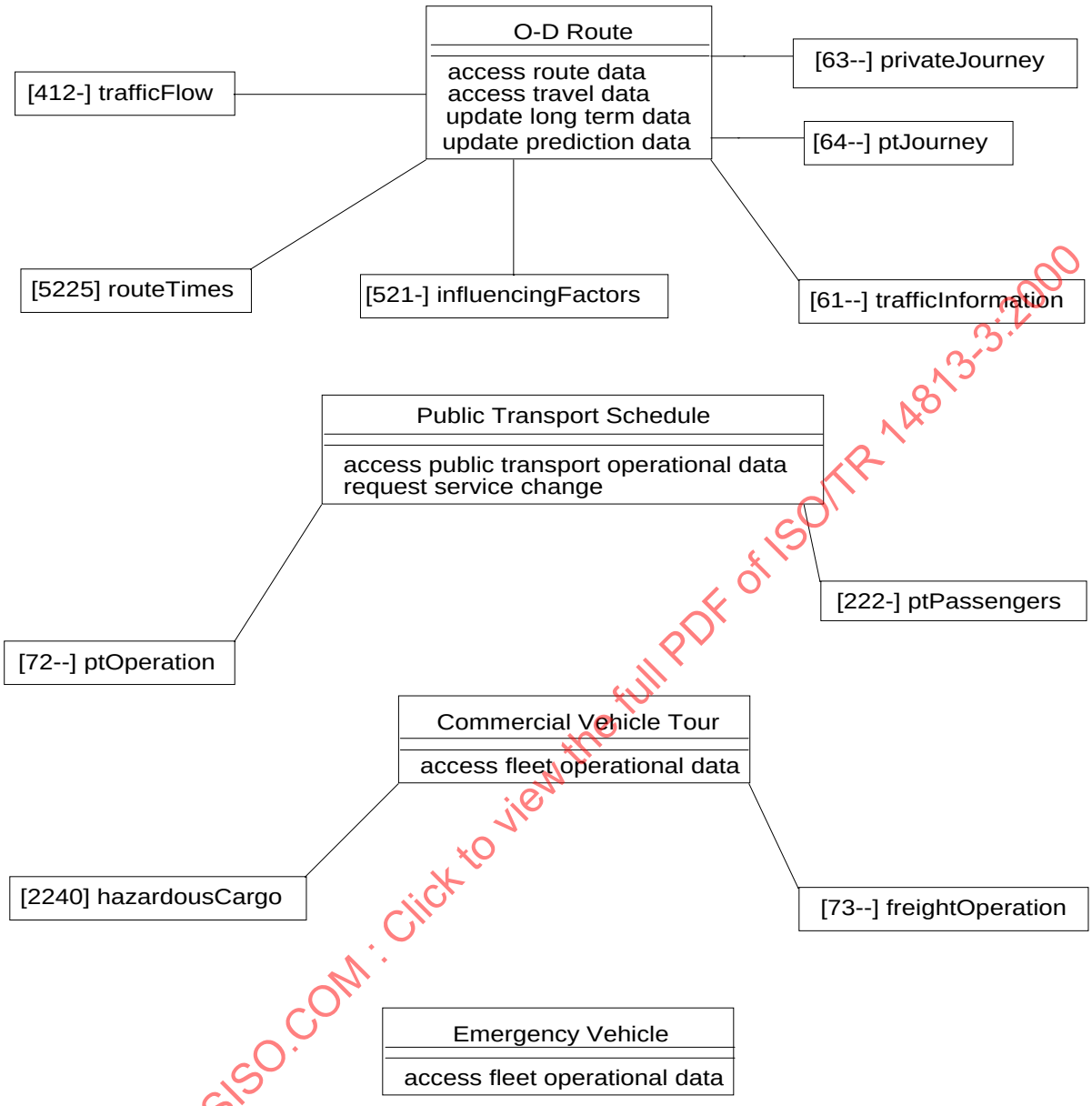

Figure 43 — Transport package class operations and information class associations for Traffic Management

7.3 Vehicle

7.3.1 Vehicle Status

The abstract operations for Vehicle Status in ISO/TR 14813 Part 2 are provided by the Vehicle Class, the Roadway Class and the Event Class. This leads to the following key classes defined in previous clauses.

— Vehicle

— Emergency

— Local Control Group

The following new interface class is required

— In-Vehicle Sensor: This interface class is invented to interface various types of sensors which are used on-board a vehicle.

The sequence diagram for one iteration is shown in Figure 44. The first stage monitors status from the vehicle:

— Vehicle.monitor driver (…)

— Vehicle.monitor vehicle operation (…)

— Vehicle.monitor roadway environment (…)

— Vehicle.monitor inter-vehicle dynamics (…)

The next stage collects data from the roadway system:

— Local Control Group.access collision avoidance data (…)

The next stage conducts the safety analysis and alerts the driver if necessary:

— Vehicle.analyse safety (…)

— Vehicle Interface.alert driver (…)

The final stage conducts the automated responses for collision avoidance:

— Vehicle.override default control mode (…)

— Vehicle Interface.get location (…)

— Emergency.create (…)

7.3.2 Vehicle Operation

The abstract operations for Vehicle Operation in ISO/TR 14813 Part 2 are provided by the Vehicle Class.

The sequence diagram is shown in Figure 45. The Driver's mode request results in the selection of one of the following macro operations:

— Vehicle.operate in manual mode (…)

— Vehicle.control vehicle actuators (…)

— Vehicle.control vehicle on automated highway (…)

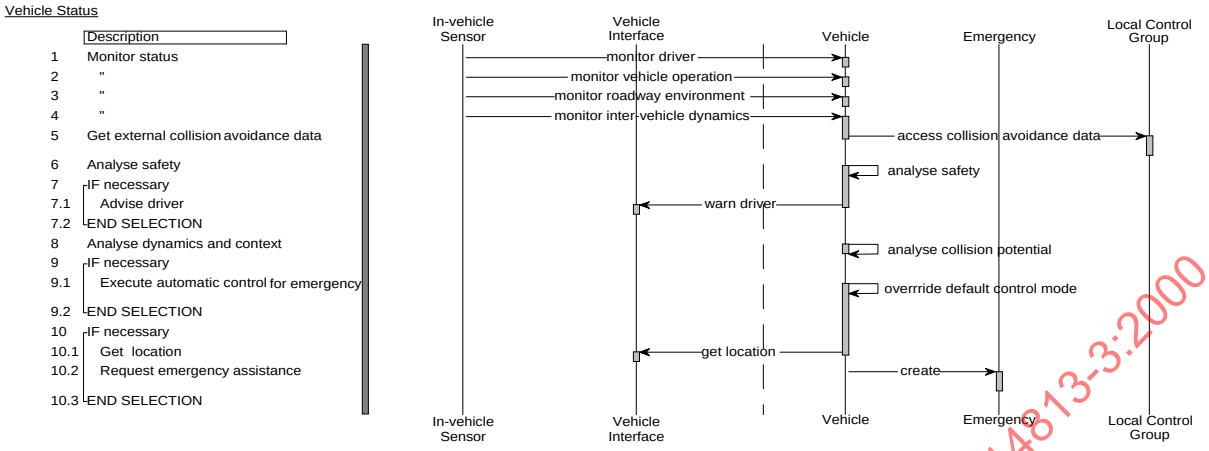

Figure 44 — Sequence diagram for Vehicle Status

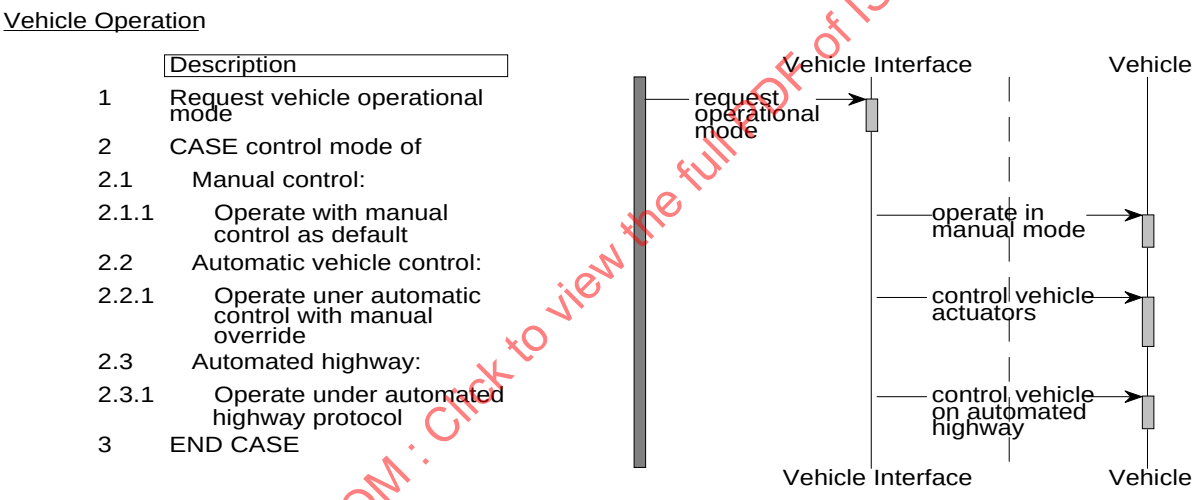

Figure 45 — Sequence diagram for Vehicle Operation

7.4 Commercial Vehicle

7.4.1 Order and Shipment

The abstract operations for Order and Shipment in ISO/TR 14813 Part 2 are provided by the Transport Class. This leads to the following key classes defined in previous clauses.

— Forwarder

— Journey

— Trip

The following new control and information class is required

— Goods Item: This class is invented to record journey planning and control the tracing of freight items.

The sequence diagram for the elements freight shipment is shown in Figure 46. The first transaction supports the selection of a forwarder from a TICS database:

— Forwarder.select forwarder(…)

The main logic is designed to support the tracing of freight. This begins with the creation of an Goods Item object:

— Goods Item.create (…)

The journey details (which have been arranged externally) are then entered into the TICS database:

— Journey.create (…)

— Journey.insert goods([223-])

Trip details are entered:

— Trip.create (…)

involving interaction with the Carrier actor

— Operating Interface.freight booking (…)

and inserting the trip details in the journey.

— Journey.insert trip(…)

As the freight movement progresses the trips are logged:

— Trip.log trip (…)

This permits the tracing of freight:

— Goods Item.trace([223-])

7.4.2 Commercial Vehicle Tour Planning

The abstract operations for Commercial Vehicle Tour Planning in ISO/TR 14813 Part 2 are provided by the Transport Class. This leads to the following key classes defined in previous clauses.

— Goods Item

— Journey

— Trip

The sequence diagram for one journey planning exercise is shown in Figure 47.The following new control and information class is required

— Freight Complement: This class is invented to register goods items in optimised cargo complements for commercial vehicles.

The first stage consists of registering goods items:

— Goods Item.log arrival ([2230])

The main stage is to arrange freight complements for tours by iterating over all goods items awaiting shipment:

— Goods Item.iterate over goods items ([2230])

— accessing the next trip for each goods item

— Goods Item.iterate over trips ([2230])

— Trip.access current trip ([2230])

— establishing a new tour if necessary

— Freight Complement.create (…)

— Journey.create (…)

— and building up the freight complement and journey list:

— Journey.insert trip (…)

— Freight Complement.update (…)

— Trip.update trip data (…)

The remainder of this sequence is accomplished for each freight complement by the transactions of Route Guidance and Navigation (see Figure 21 at line 2.1), followed by transactions of Commercial Vehicle Administrative Processes (7.4.3):

7.4.3 Commercial Vehicle Administrative Processes

The abstract operations for Commercial Vehicle Administrative Processes ISO/TR 14813 Part 2 are provided by the Transport Class and the Payment Class. This leads to the following key classes defined in previous clauses.

— Commercial Vehicle

— Carrier

— Journey

— Journey Schedule

The following new control and information classes are required

— Commercial Vehicle Driver: An information class used to maintain data about commercial drivers.

— Commercial Vehicle Tour: This class is invented to control journey planning and execution for commercial vehicles. It is a specialisation of Journey.

— Tax or Fee: An information class which maintains the various charges which are incurred for use of Roadway Resources.

The sequence diagram for one commercial vehicle tour administrative exercise is shown in Figure 48. The first stage is for clearance:

— Commercial Vehicle Tour.add driver and carrier ([221-])

— Carrier.clear carrier (…)

— Commercial Vehicle.clear vehicle ([2111])

— Commercial Vehicle Driver.clear driver ([221-])

followed by the conformance agency checks:

— Operating Interface.conformance agency check (…)

The second stage is for pre-payment:

— Commercial Vehicle Tour.update credit identity ([2214])

— Journey.iterate over trips (…)

— Tax or Fee.access segment charges (…)

— Journey Schedule.update time confirmation and price (…)

followed by the transactions of Journey Payment.

7.4.4 Commercial Vehicle Road Operation

The abstract operations for Commercial Vehicle Road Operation in ISO/TR 14813 Part 2 are provided by the Transport Class, the Roadway Class and the Event Class. This leads to the following key classes defined in previous clauses.

— Resource

— Commercial Vehicle Tour

— Commercial Vehicle

— Violation

The sequence diagram for the different stages of vehicle operation during a commercial vehicle tour is shown in Figure 49. The first stage is to initialise the vehicle interface from data already associated with an object of the Commercial Vehicle Tour class:

— Commercial Vehicle Tour.initiate vehicle interface (…)

The next stage comprises the iteration which monitors the state of the vehicle and its cargo:

— In-vehicle sensor.monitor vehicle and cargo ([214-])

— Commercial Vehicle Tour.update vehicle and cargo data (…)

The next stage comprises the iteration which allows the Carrier Operator to instruct the driver:

— Commercial Vehicle Tour.interrogate on-board data([214-])

— Commercial Vehicle Tour.update driver instructions([6620])

— Special Vehicle Interface.driver action request ([6620])

The next stage is about route guidance and other information using On-trip Traveller Information (Figure 24 line 3).

The next stage is the iteration repeated for each resource usage:

— Resource.recognise vehicle([41--])

— Commercial Vehicle.classify vehicle([41-])

— Resource.clear vehicle([211-])

— Special Vehicle Interface.interrogate on-board data (…)

— Commercial Vehicle Tour.access trip data (…)

— Commercial Vehicle Tour.command vehicle ([612-])

— Special Vehicle Interface.request driver action ([612-])

— Violation.create {[412-])

— Commercial Vehicle Tour.log trip segment (…)

The final stage is used to process any violation and notify the conformance agency:

— Violation.process violation ([811-])

— Operating Interface.report violation (([811-])

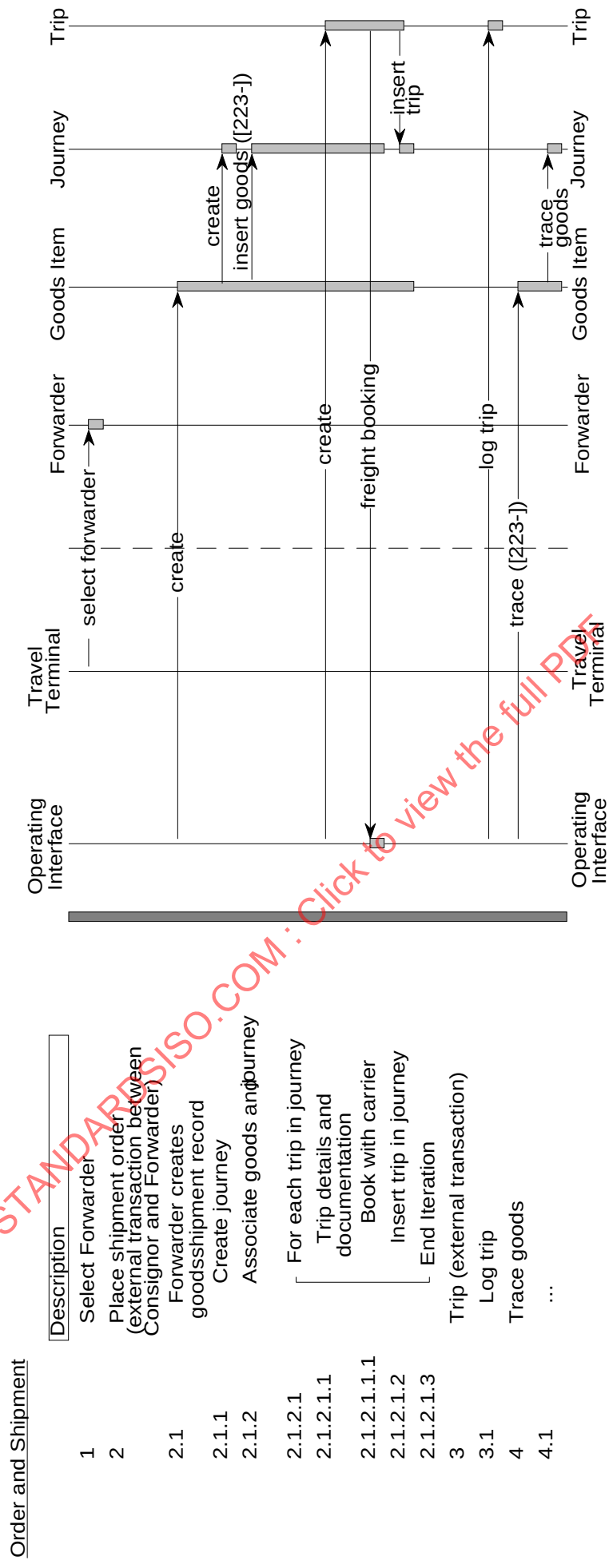

Figure 46 — Sequence diagram for Order and Shipment

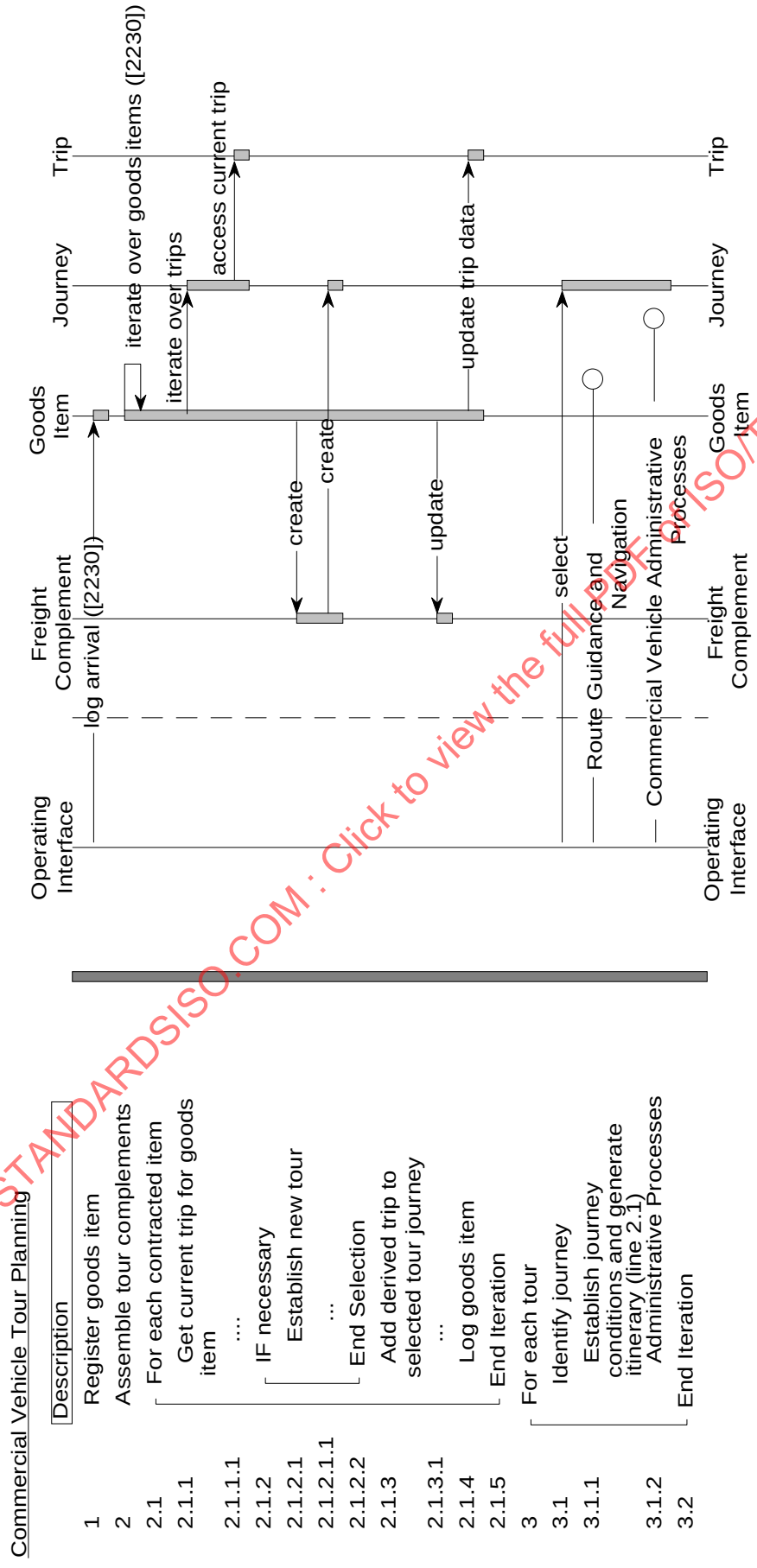

Figure 47 — Sequence diagram for Commercial Vehicle Tour Planning

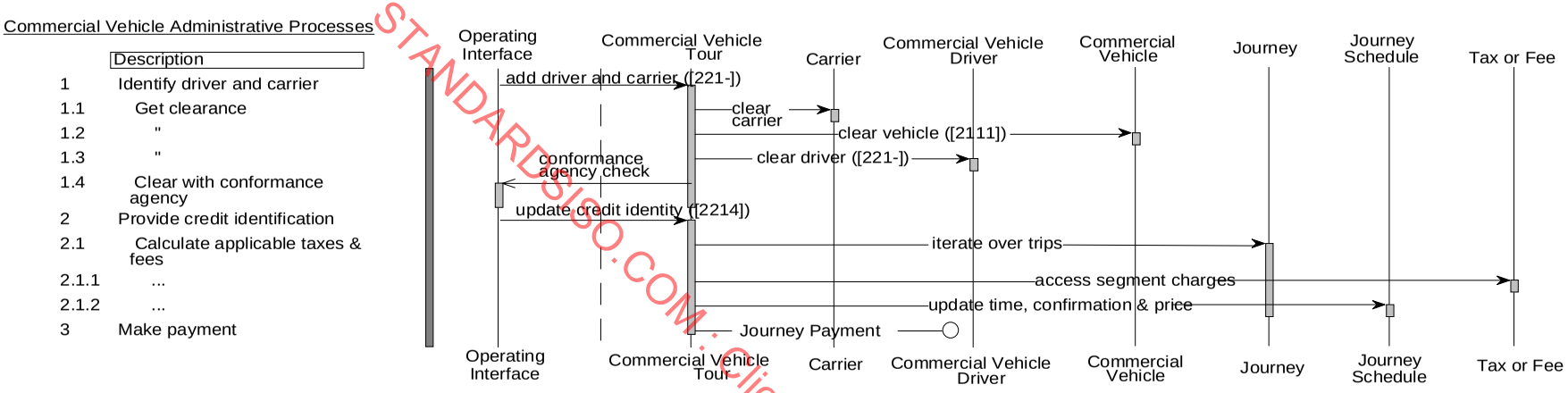

Figure 48 — Sequence diagram for Commercial Vehicle Administrative Processes

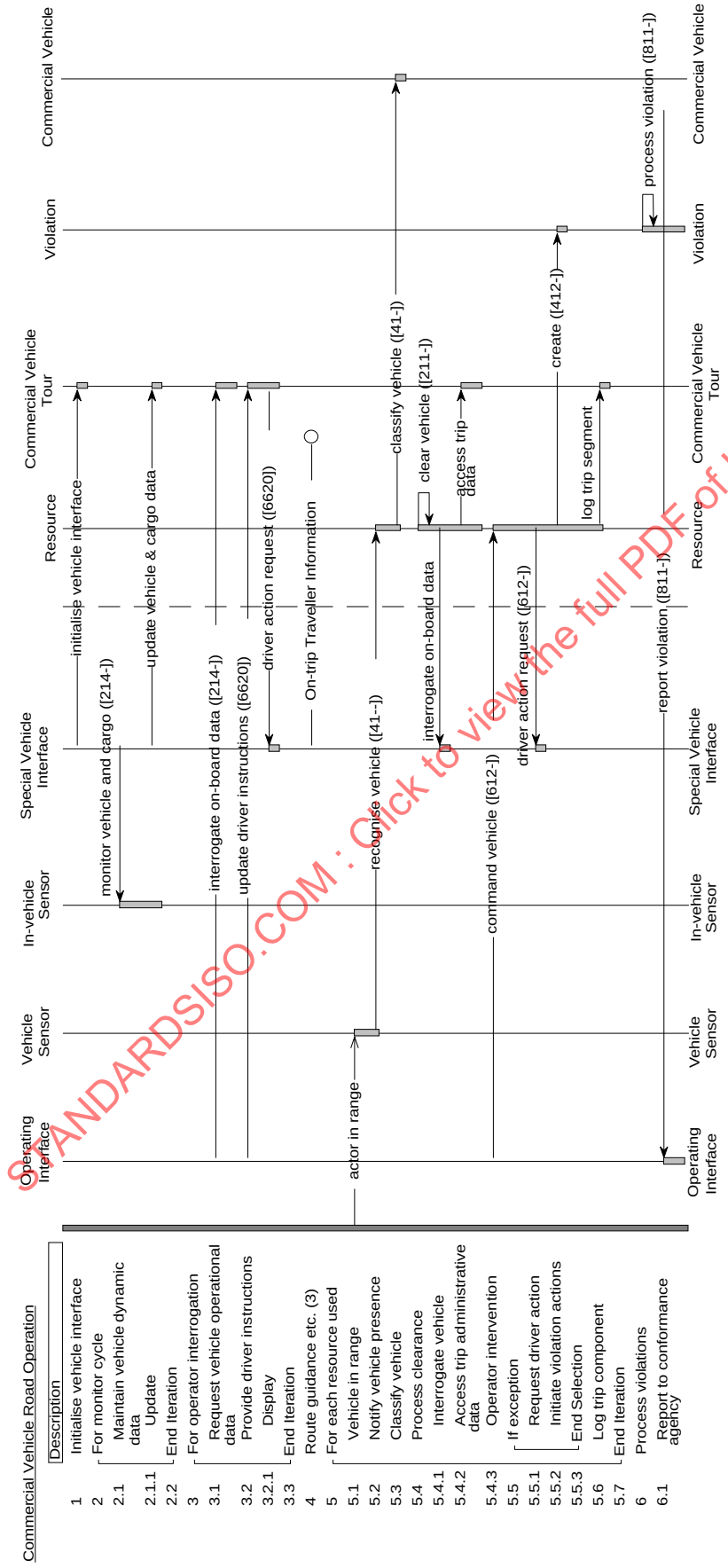

Figure 49 — Sequence diagram for Commercial Vehicle Road Operation

7.5 Public Transport

7.5.1 Route and Schedule Planning

The abstract operations for Route and Schedule Planning in ISO/TR 14813 Part 2 are provided by the Transport Class. This leads to the following key classes defined in previous clauses.

— Public Transport Route

— Public Transport Schedule

— Roadway Group

— O-D Route

The sequence diagram is shown in Figure 50. The first stage maintains the geographic database:

— Public Transport Route.update map data ([121-])

The remainder of the sequence is an iteration conducted for each route. There are two cases, the first for maintenance of an existing route:

— Public Transport Schedule.request service change (…)

— Public Transport Schedule.update schedule (…)

— Special Vehicle Interface.access vehicle and passenger data([222-],[722-])

— Public Transport Route.access route([121-])

— O-D Route.access current conditions ([611-])

The second case is for route planning:

— Public Transport Route.create([121-])

— Public Transport Route.update route([121-])

— Public Transport Schedule.integrate vehicle and passenger data (…)

— O-D Route.access route data ([522-],[7---])

— Public Transport Schedule.update schedule (…)

The remaining logic is to generate and publish the timetable:

— Public Transport Schedule.generate timetable (…)

— Travel Terminal.publish timetable ([7211])

7.5.2 Fixed Route Public Transport

The abstract operations for Fixed Route Public Transport in ISO/TR 14813 Part 2 are provided by the Transport Class and the Roadway Class. This leads to the following key classes defined in previous clauses.

— Public Transport Schedule

— Roadway Group

The sequence diagram for operational maintenance of fixed route services is shown in Figure 51. The first stage consists of accumulating all operational data and updating the schedule:

— Public Transport Schedule.update intermodal connection data (…)

— Public Transport Schedule.update schedule (…)

— Roadway Group.access prediction data([522-])

— Roadway Group.access public transport priority data(…)

— Special Vehicle Interface.access vehicle and passenger data([222-])

— Roadway Group.priority request (…)

The next stage accomodates operator overrides::

— Public Transport Schedule.update schedule (…)

— Roadway Group.priority request (…)

The final stages are to instruct drivers and to support traveller information enquiries:

— Special Vehicle Interface.update driver instructions(…)

— Public Transport Schedule.access public transport running data ([72--])

7.5.3 Demand Responsive Public Transport

The abstract operations for Demand Responsive Public Transport in ISO/TR 14813 Part 2 are provided by the Transport Class. This leads to the following key classes defined in previous clauses.

— Journey

The following new control and information classes are required:

— Demand Responsive Vehicle: This class is used for dynamic dispatching of the fleet.

— Demand Responsive Vehicle Itinerary: This class is used for dynamic scheduling of the fleet.

The sequence diagram for a reservation request is shown in Figure 52. The first stage is to support traveller information ad hoc enquiries:

— Demand Responsive Vehicle Itinerary.reservation enquiry (…)

The main stage supports fleet scheduling and vehicle allocation:

— Demand Responsive Vehicle Itinerary.make reservation (…)

— Special Vehicle Interface.interrogate on-board data(…)

The last operation involves the Location Data Source actor. If a new vehicle has to be dispatched:

— Demand Responsive Vehicle.deploy (…)

— Special Vehicle Interface.update driver instructions(…)

The operator may override the dispatch decision:

— Operating Interface.operator data output(…)

— Demand Responsive Vehicle Itinerary.operator assignment (…)

The scheduling decision is finalised:

— Demand Responsive Vehicle.update passenger data (…)

— Special Vehicle Interface.update driver instructions(…)

with route guidance being provided through Route Guidance and Navigation (Figure 21 line 2.2).

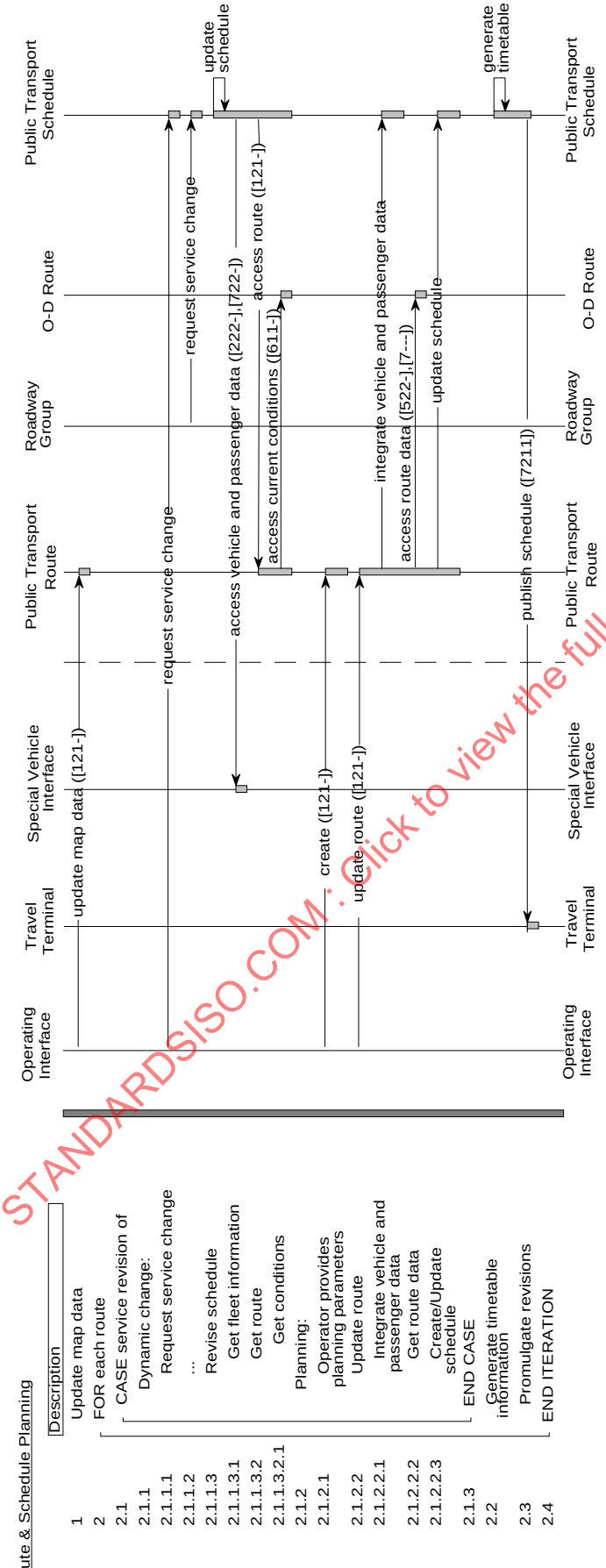

Figure 50 — Sequence diagram for Route and Schedule Planning

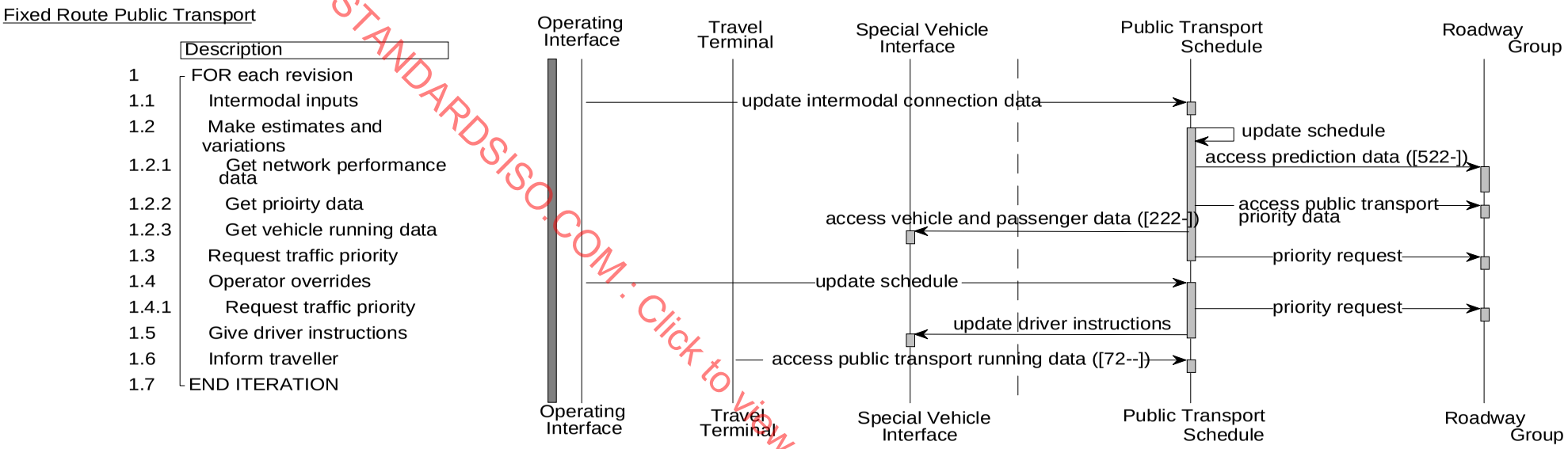

Figure 51 — Sequence diagram for Fixed Route Public Transport

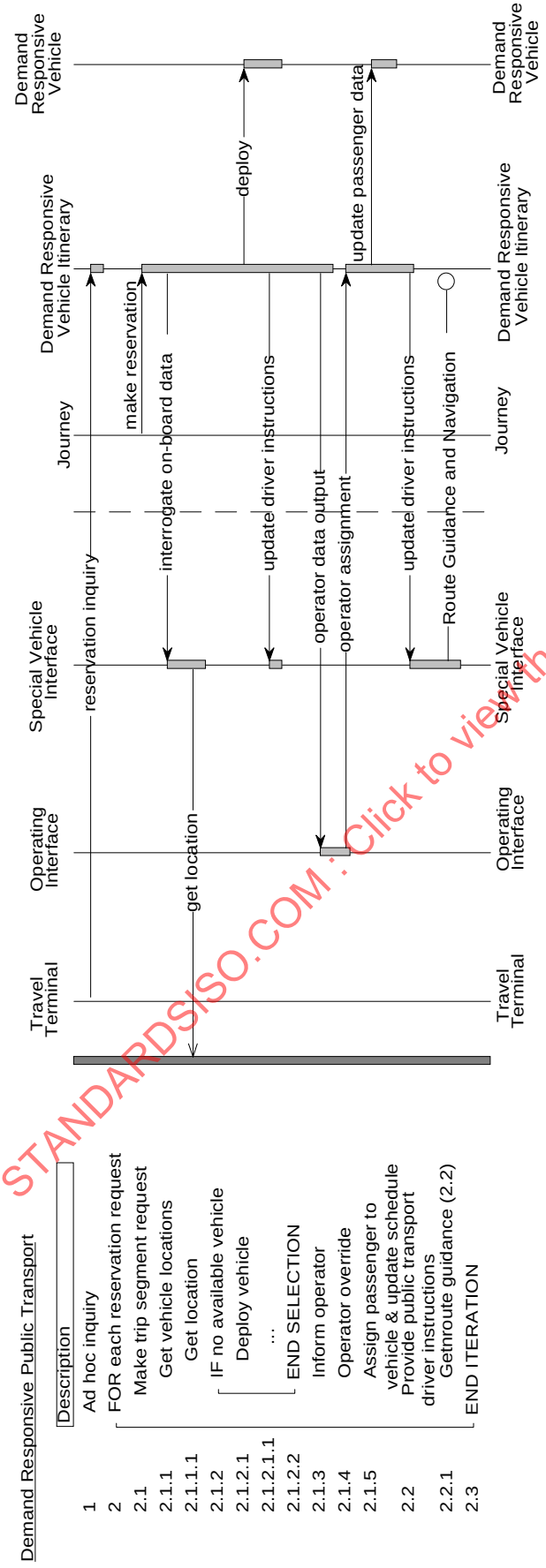

Figure 52 — Sequence Diagram for Demand Responsive Public Transport

7.6 Emergency

7.6.1 Emergency Notification and Personal Security

The abstract operations for Emergency Notification and Personal Security in ISO/TR 14813 Part 2 are provided by the Event Class. This leads to the following key class defined in previous clauses.

— Emergency

The sequence diagram for one notification is shown in Figure 53. The first stage is to validate the Traveller's journey request:

— Travel Terminal**.**sense security violation (…)

— Travel Terminal**.**security request (…)

These operations may involve the Location Data Source actor.

An emergency object is created:

— Emergency**.**create (…)

and the Emergency Resources Allocation transactions (Figure 54 line 2.2) invoked.

Finally the request is acknowledged:

— Travel Terminal**.**acknowledge (…)

7.6.2 Emergency Resources Allocation

The abstract operations for Emergency Resources Allocation in ISO/TR 14813 Part 2 are provided by the Event Class. This leads to the following key classes defined in previous clauses.

— Emergency

— Roadway Group

— Emergency Vehicle

The following new information class is required

— Emergency Response: This information class is invented to maintain response plans to be used in emergency management.

The sequence diagram is shown in Figure 54. The first stage involves the maintenance and planning operations:

— Emergency**.**update map data ([131-])

— Emergency Response**.**create ([5313])

The main stage registers an emergency and determine if a response is required:

— Emergency**.**create ([44--])

— Emergency**.**incident or emergency message ([54--])

— Emergency**.**data fusion (…)

— Emergency.assess status (…)

— Travel Terminal.acknowledge (…)

The second part of the main stage determines the response, allowing operator override:

— Emergency Response.match emergency ([44--],[5313])

— Emergency.update response plan ([5313])

Resources are deployed by the transactions of Emergency Vehicle Management (7.6.3). The incident is notified if necessary:

— Roadway Group.update incident data[44--])

The final stage is to integrate the state of current emergenies:

— Emergency.exchange data (…)

7.6.3 Emergency Vehicle Management

The abstract operations for Emergency Vehicle Management in ISO/TR 14813 Part 2 are provided by the Transport Class and the Roadway Class. This leads to the following key classes defined in previous clauses.

— Emergency

— Emergency Response

— Emergency Vehicle

— Roadway Group

The sequence diagram is shown in Figure 55. The first stage allows a response plan to be applied:

— Emergency.apply response plan (…)

— Emergency Response.access response plan ([5313])

— Operating Interface.operator data output (…)

— Emergency Vehicle.deploy (…)

— Special Vehicle Interface.update driver instructions(…)

Once the vehicles are committed the transactions of Route Guidance and Navigation (Figure 21 line 2.1) are used and then any traffic priority requested:

— Roadway Group.priority request(…)

The final part of the sequence is to iteratively assess the status of the emergency:

— Emergency.assess status (…)

— Special Vehicle Interface.track vehicle and status (…)

This operation may involve the Location Data Source actor.

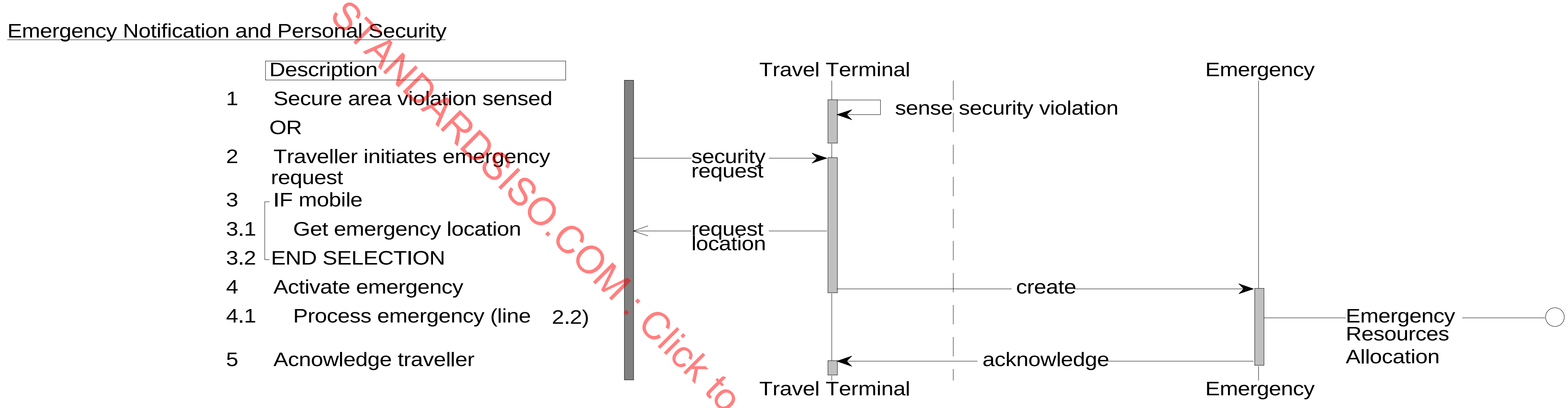

Figure 53 — Sequence diagram for Emergency Notification and Personal Security

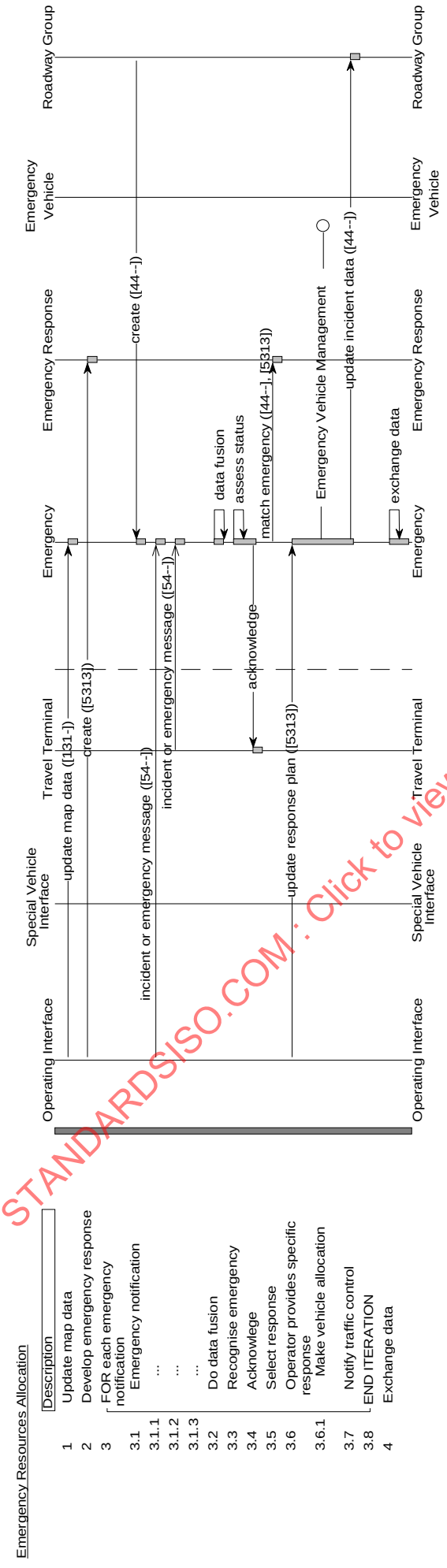

Figure 54 — Sequence diagram for Emergency Resources Allocation

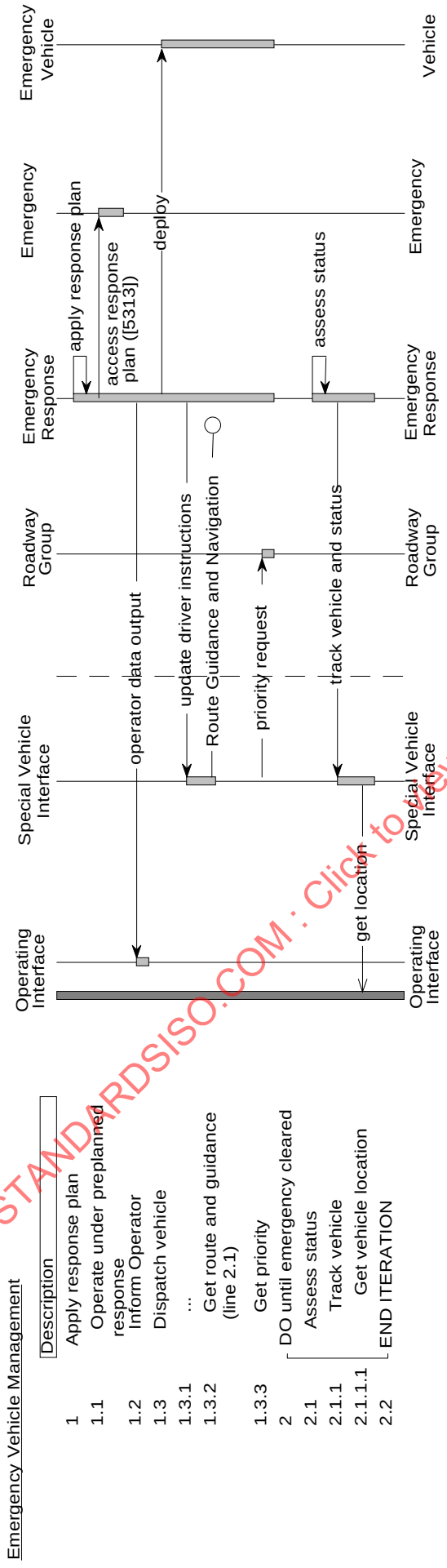

Figure 55 — Sequence diagram for Emergency Vehicle Management

7.7 Electronic Payment

7.7.1 Payment Means

The abstract operations for Payment Means in ISO/TR 14813 Part 2 are provided by the Roadway Class. This leads to the following key classes defined in previous clauses.

— Resource

The sequence diagram is shown in Figure 56. There are two independent stages. The first stage consists of the standard transactions for priming an electronic purse:

— Travel Terminal.read payment card/tag ([2214])

— Travel Terminal.accept cash/credit (…)

This is followed by a message to the Collection Agent actor. The second stage is for establishing credit for use of Resources:

— Travel Terminal.accept cash/credit (…)

— Resource.validate user credit (…)

This is followed by a message to the Clearing Agent actor.

— Operating Interface.register user credit (…)

7.7.2 Fare Collection

The abstract operations for Fare Collection in ISO/TR 14813 Part 2 are provided by the Payment Class. This leads to the following key classes defined in previous clauses.

— Billing Record

The sequence diagram is shown in Figure 57. Most of the interactions are in the interface classes:

— In-vehicle sensor.detect passenger embark/disembark (…)

— Special Vehicle Interface.count passengers (…)

— Travel Terminal.read payment card/tag ([2214])

— Billing Record.create([2214])

This is followed by the Payment Transaction (Figure 59 line 4.3).

7.7.3 Vehicle Charges

The abstract operations for Vehicle Charges in ISO/TR 14813 Part 2 are provided by the Roadway Class and the Payment Class. This leads to the following key classes defined in previous clauses.

— Resource

— Billing Record

The sequence diagram for one journey planning exercise is shown in Figure 58. The first stage is to establish the payment means:

— Vehicle Interface. read payment card/tag ([2214])

The next stage involves recognising a vehicle at the resource:

— Vehicle Sensor.specific actor in range

— Resource.recognise vehicle([411-])

followed by retrieving the identification and payment means from the interface objects:

— Vehicle Interface.access payment identification ([2214])

— Vehicle Sensor.access vehicle identification ([4111])

If necessary the driver is warned:

— Vehicle Interface.warn driver ([6132])

then the payment process begins:

— Billing Record.create([4112],[2214])

followed by the Payment Transaction (Figure 59 line 2).

7.7.4 Payment Transaction

The abstract operations for Payment Transaction in ISO/TR 14813 Part 2 are provided by the Payment Class and the Event Class. This leads to the following key classes defined in previous clauses.

— Resource

— Billing Record

— Tariff

— Probe

— Violation

The following new information class is required

— Advance Payment: This information class is invented to apply advance payments made for Resource use.

The sequence diagram comprising all the possible stages is shown in Figure 59. The first stage is a maintenance stage for tariffs:

— Tariff.update tariff ([715-])

If the transactions are being invoked at the completion of a Resource use, e.g. Vehicle Charges (7.7.3), the Billing Record object has to be selected, Probe data recorded, and any invalid use acted upon:

— Billing Record.select ([2214])

— Probe.update resource use statistics (…)

— Violation.create([211-])

The next stage computes charges in various circumstances:

— Billing Record.enter completion details ([2214])

— Tariff.calculate charges (…)

— Advance Payment.reconcile[2214])

if credit

— Billing Record.record creditor amount (…)

if debit

— Travel Terminal.debit request (…)

which results in an electronic purse calculation or the acceptance of cash or token. Again violations are processed:

— Violation.create([2214])

Advance payments are processed:

— Advance Payment.create ([2214])

Periodically all payments are reconciled and request sent to the Clearing Operator agents:

— Billing Record.reconcile collected fees (…)

— Operating Interface.request funds transfer ([8210],[8300],[8400],[8500])

The final stage is used to process any violation and notify the conformance agency:

— Violation.process violation (…)

— Operating Interface.report violation (([811-])

Payment Means

	Description
1	IF pre-payment means
1.1	Provide payment instrument
1.2	Provide cash/credit
1.3	Collection agent transaction
1.4	End Selection
2	IF post-payment means
2.1	Provide credit identification
2.2	Validate credit
2.2.1	Send user/issuer data to Clearing Operator
2.3	End Selection

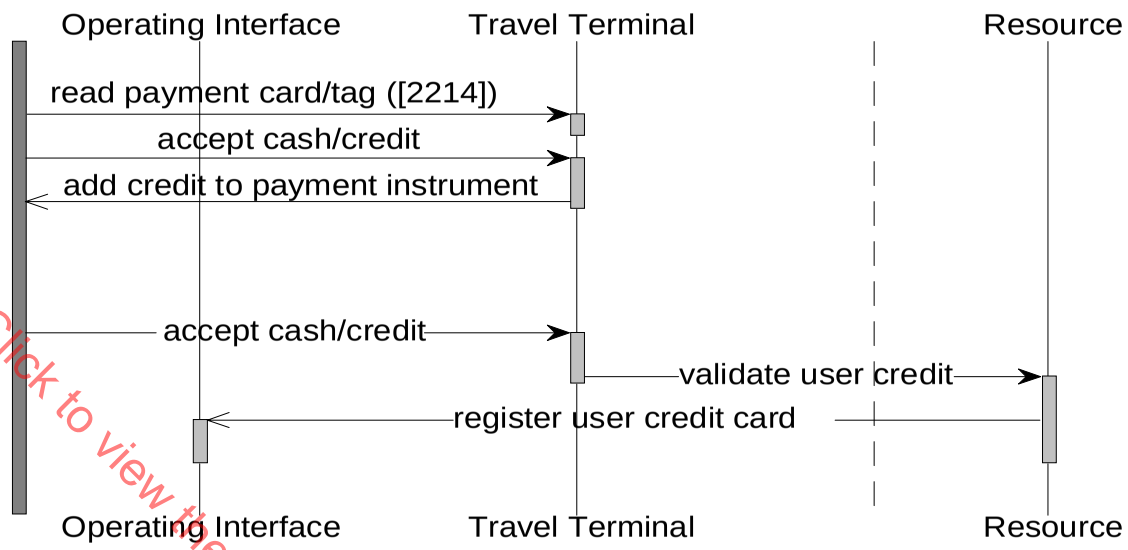

Figure 56 — Sequence diagram for Payment Means

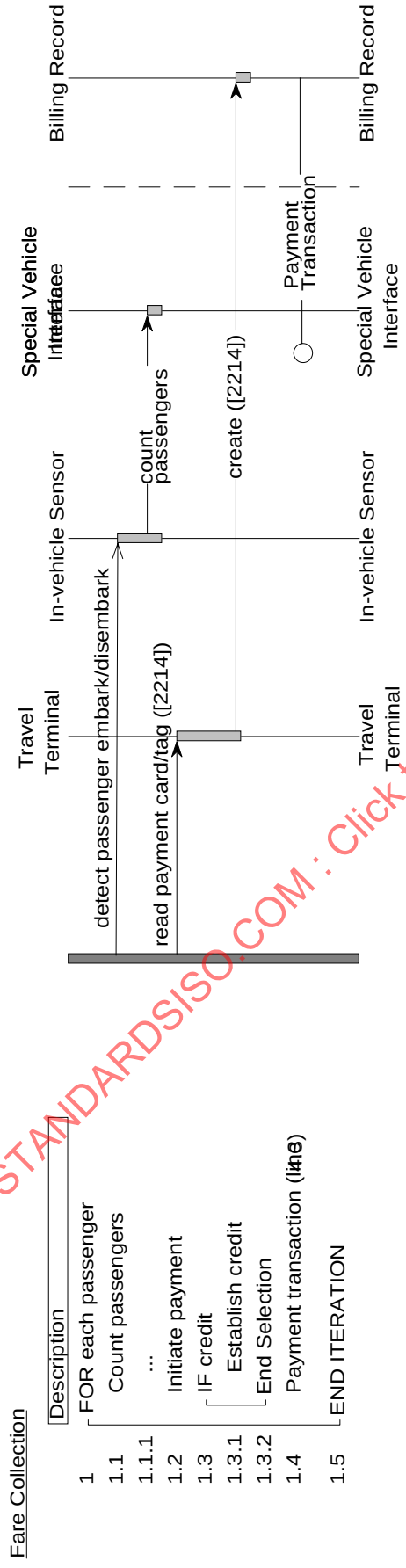

Figure 57 — Sequence diagram for Fare Collection

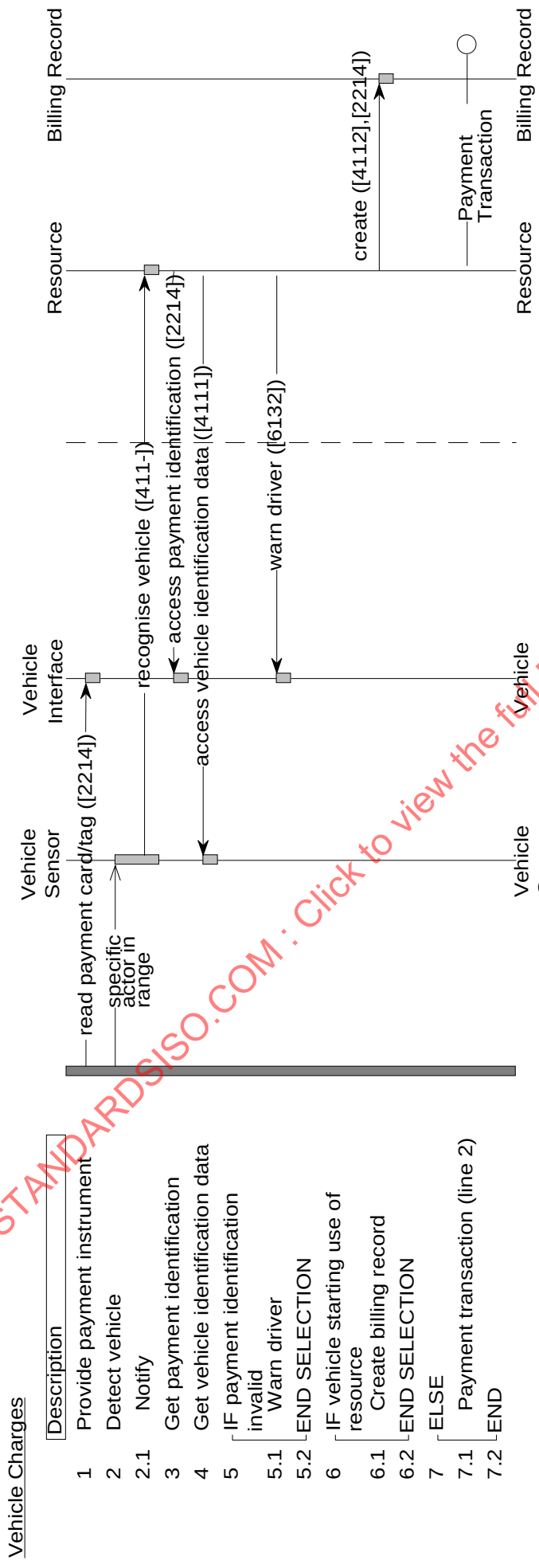

Figure 58 — Sequence diagram for Vehicle Charges

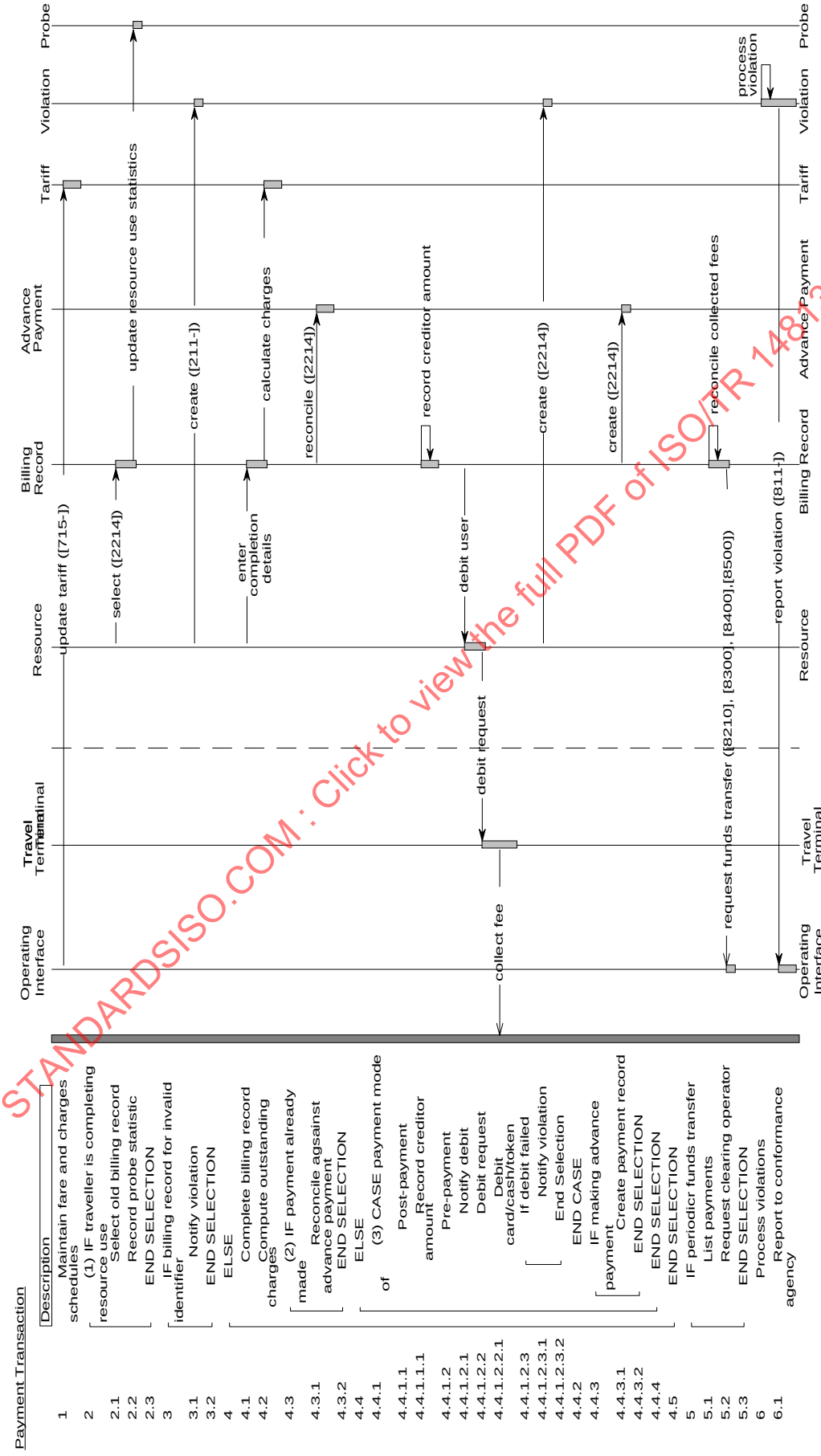

Figure 59 — Sequence diagram for Payment Transaction

7.8 Safety

7.8.1 Safety Enhancement for Vulnerable Road Users

The abstract operations for Safety Enhancement for Vulnerable Road Users in ISO/TR 14813 Part 2 are provided by the Roadway Class and the Vehicle Class. This leads to the following key classes defined in previous clauses.

— Local Control Group

— Vehicle

The sequence diagram is shown in Figure 60. It is a continual iteration. In the first stage the movements of vehicles and pedestrians are continually monitored and appropriate warnings conveyed:

— Local Control Group**.**identify vulnerable (…)

— Vehicle Interface**.**warn driver ([612-])

— Roadside Peripheral**.**safety warning (…)

Another stage focuses on vulnerable vehicles:

— Vehicle**.**monitor vulnerable vehilces(…)

— Vehicle Interface**.**warn driver ([612-])

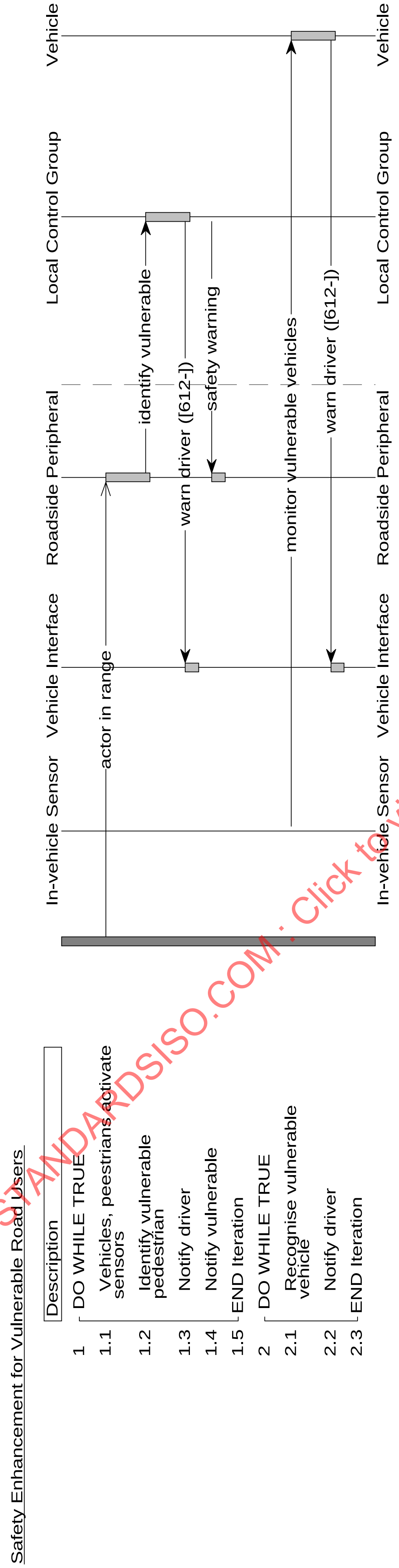

Figure 60 — Sequence diagram for Safety Enhancement for Vulnerable Road Users

8 Elaboration of the Packages

This clause summarises the packages by developing a class diagram containing the key classes of the package and their associations. The operations of each class are then listed . In 8.7 a matrix relating classes to their involvement in sequence diagrams (hence use case) is given.

8.1 Roadway

The class diagram for the key classes of the Roadway package is shown in Figure 61.

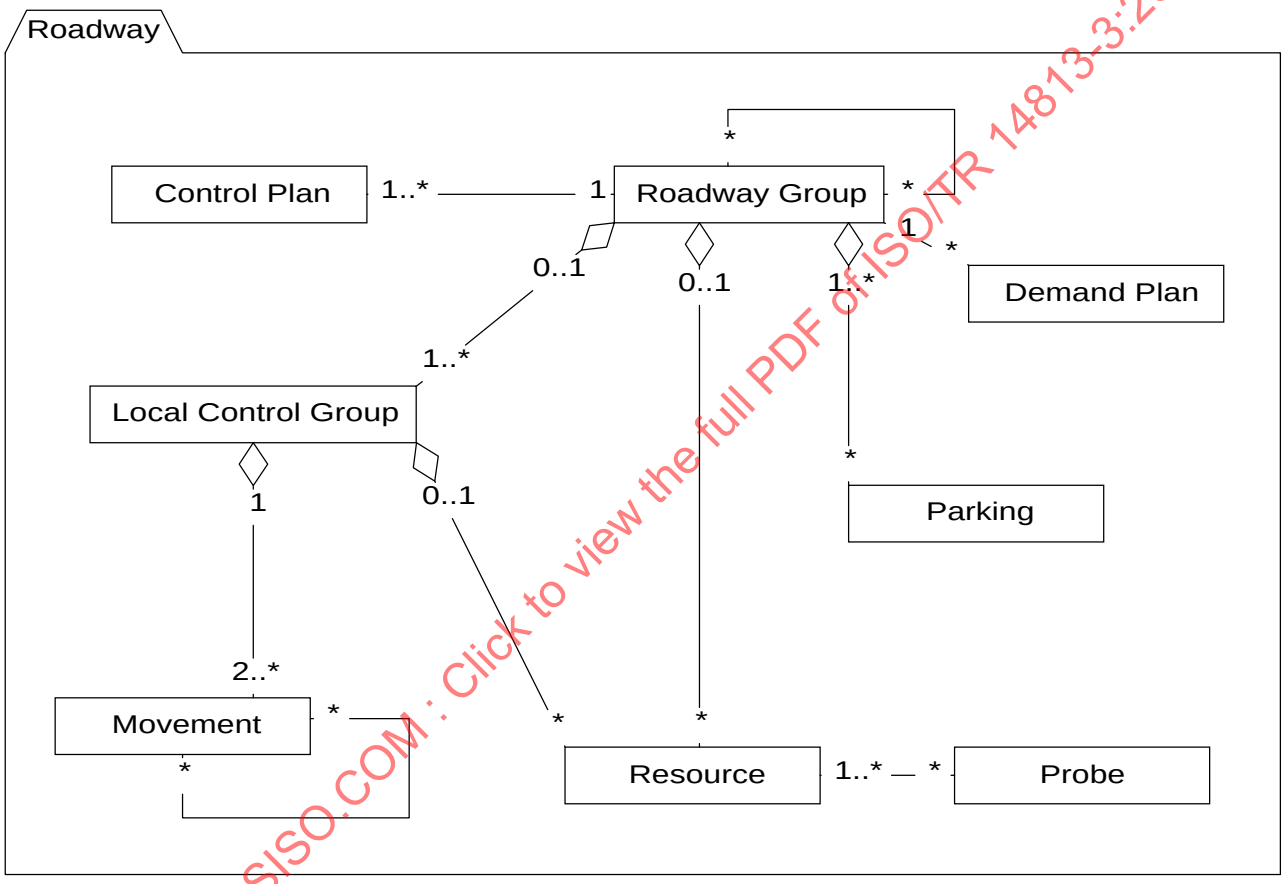

Figure 61 — Key classes of the Roadway package

Control Plan

This class has the following operations:

os+[3] create

o+ compute plan

Demand Plan

3) os+ denotes a static operation (see ISO 14813 Part 4)

This class has the following operations:

os+ create

o+ compute plan

o+ update demand policy data

Local Control Group

This class has the following operations:

o+[4] access collision avoidance data

o+ apply traffic regulations

o+ generate control parameters for peripherals

o+ identify vulnerable

o+ priority request

o+ update control data

o+ update rail traffic data

o+ update sensor measurements

Movement

This class has the following operations:

o+ update map data

o+ update measurements

Parking

This class has the following operations:

o+ access parking data

o+ make reservation

o+ request to cater to demand

o+ update control data

o+ update long term data

o+ update status

Probe

This class has the following operations:

os+ access route travel time data

o+ aggregate statistics

o+ update resource use statistics

[4] o+ denotes an instance (object) operation (see ISO 14813 Part 4).

Resource

This class has the following operations:

- o+ clear vehicle
- o+ command vehicle
- os+ create
- o+ debit user
- o+ recognise vehicle
- o+ validate user credit

Roadway Group

This class has the following operations:

- o+ access prediction data
- o+ access public transport priority data
- o+ access traffic data
- o+ analyse data
- o+ data fusion
- o+ decision support for strategy development
- os+ exchange data
- o+ exchange predictions
- o+ integrate measurements
- o+ predict demand
- os+ predict network performance
- o+ priority request
- o+ schedule maintenance
- os+ compute strategy
- o+ support operator access
- o+ transport planner request
- o+ update control strategy
- o+ update current data
- o+ update demand strategy
- o+ update incident data
- o+ update incident strategy
- o+ update long term data
- o+ update map data
- o+ update performance data
- o+ update static data
- o+ update traffic control data

8.2 Transport

The class diagram for some of the key classes of the Transport package is shown in Figure 62. More key classes of the Transport package are shown in association with the Vehicles package.

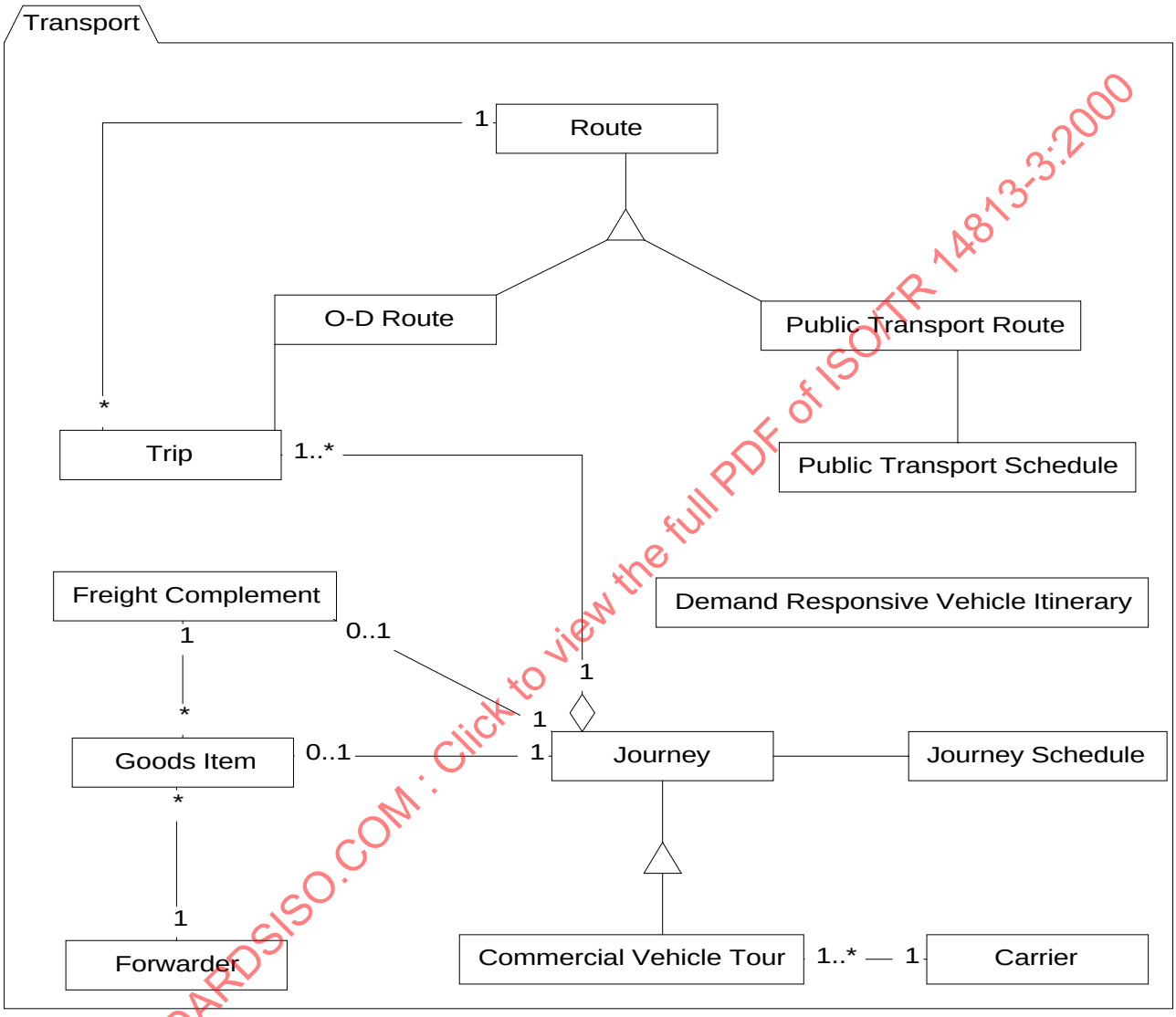

Figure 62 — Key classes of the Transport package

Carrier

This class has the following operations:

o+ clear carrier

Commercial Vehicle Tour

This class has the following operations:

os+ access fleet operational data
o+ access trip data
o+ add driver and carrier
o+ initialise vehicle interface
o+ interrogate on-board data
o+ log trip segment
o+ update credit identity
o+ update driver instructions
o+ update vehicle & cargo data

Demand Responsive Vehicle Itinerary

This class has the following operations:
os+ make reservation
o+ operator assignment
os+ reservation inquiry

Forwarder

This class has the following operations:
os+ select forwarder

Freight Complement

This class has the following operations:
os+ create
o+ update

Goods Item

This class has the following operations:
os+ create
os+ iterate over goods items
o+ log arrival
o+ trace

Journey

This class has the following operations:
o+ commit
os+ create

o+ insert goods
o+ insert trip
o+ iterate over trips
o+ select
o+ trace goods
os+ update map data

Journey Schedule

This class has the following operations:
o+ access schedule
o+ access segment and cost
os+ create
o+ update time, confirmation & price
o+ validate

O-D Route

This class has the following operations:
o+ access advisory data
o+ access current conditions
o+ access route conditions
o+ access route data
o+ access route definition
os+ access travel data
o+ exchange data
o+ generate advice
o+ update effects data
o+ update long term data
o+ update map data
o+ update prediction data
o+ update travel data

Trip

This class has the following operations:
o+ access current trip
o+ access route
o+ access route guidance
os+ create

o+ generate route & guidance

o+ log trip

o+ update trip data

Public Transport Route

This class has the following operations:

o+ access route

os+ create

o+ update map data

o+ update route

Public Transport Schedule

This class has the following operations:

os+ access public transport operational data

o+ access public transport running data

o+ generate timetable

os+ integrate vehicle and passenger data

o+ request service change

o+ update intermodal connection data

o+ update schedule

8.3 Vehicles

The class diagram for the key classes of the Vehicles package is shown in Figure 63.

Commercial Vehicle

This class has the following operations:

os+ access fleet operational data

os+ classify vehicle

o+ clear vehicle

o+ log trip segment

Commercial Vehicle Driver

This class has the following operations:

o+ clear driver

Demand Responsive Vehicle

This class has the following operations:

Public Transport Vehicle...

o+ update passenger data

Special Vehicle...

o+ access fleet operational data

os+ deploy

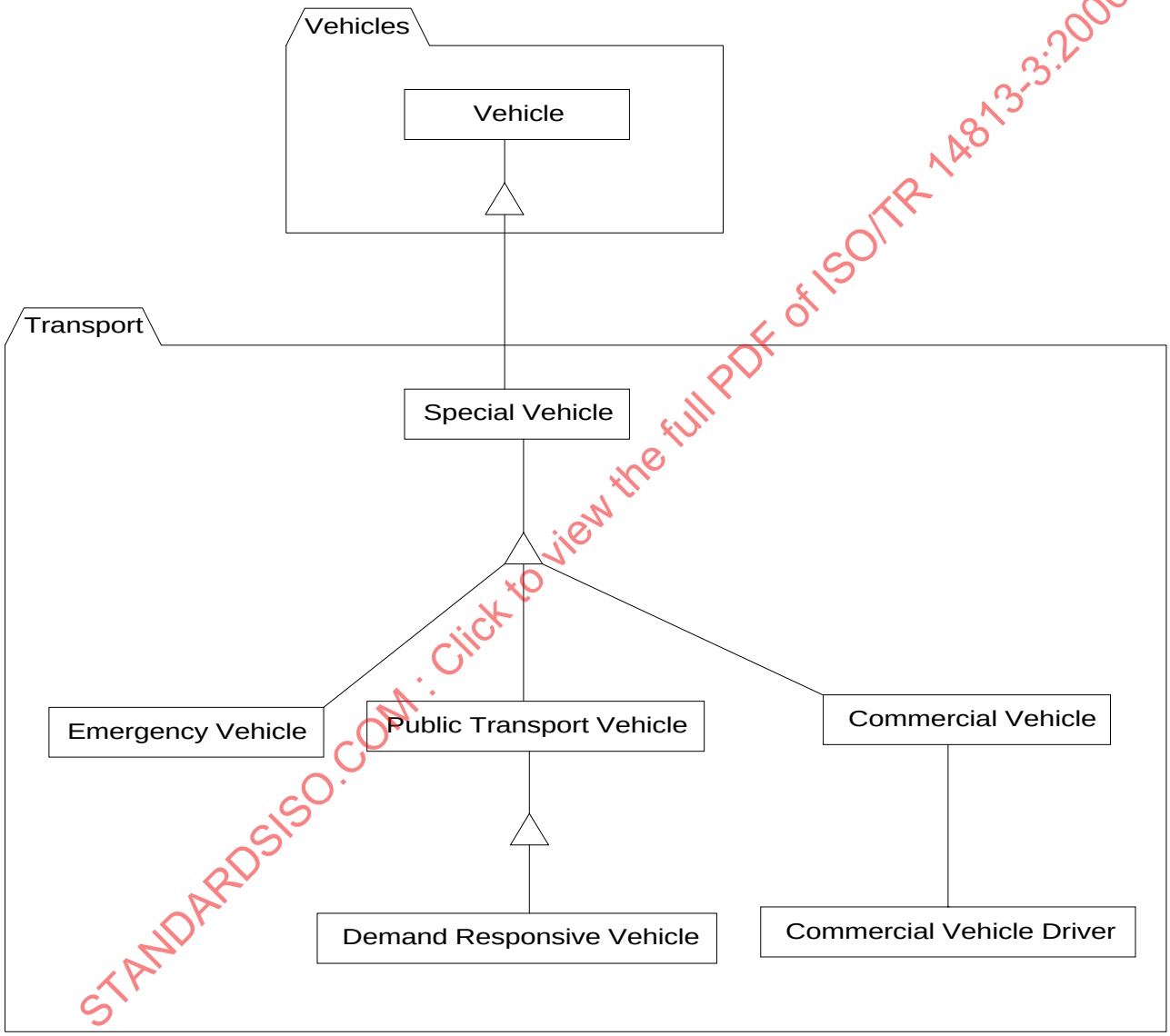

Figure 63 — Key classes of the Vehicle package

Emergency Vehicle

This class has the following operations:

Special Vehicle...

o+ access fleet operational data

os+ deploy

Public Transport Vehicle

This class has the following operations:
o+ update passenger data

Special Vehicle

This class has the following operations:
o+ access fleet operational data
os+ deploy

8.4 Events

The class diagram for the key classes of the Events package is shown in Figure 64.

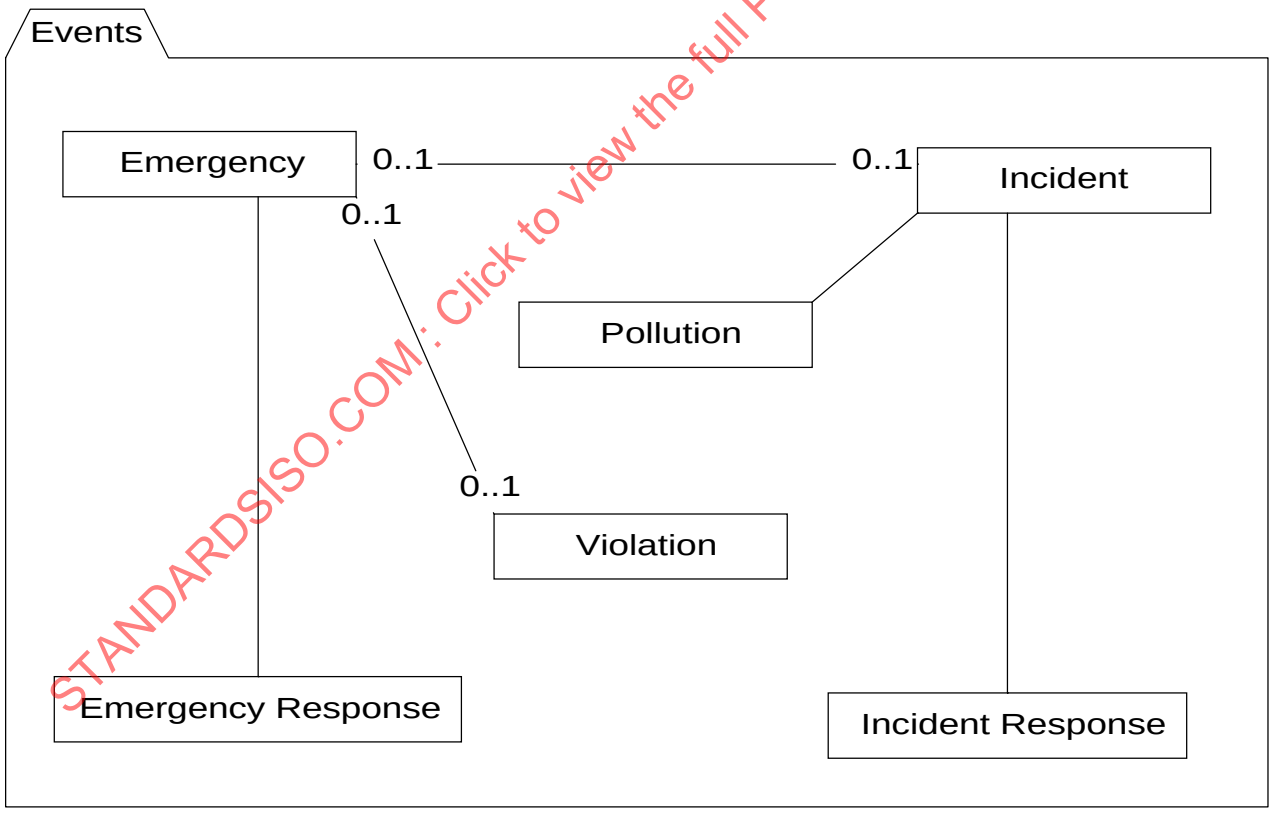

Figure 64 — Key classes of the Events package

Emergency

This class has the following operations:

os+ access fleet operational data

o+ apply response plan

o+ assess status

os+ create

o+ data fusion

o+ exchange data

os+ incident or emergency message

os+ update map data

o+ update response plan

Emergency Response

This class has the following operations:

o+ access response plan

os+ create

o+ match emergency

Incident

This class has the following operations:

os+ access incident data

os+ access predicted incidents

os+ create

o+ update long term data

o+ update response data

Incident Response

This class has the following operations:

os+ create

os+ match incident

Pollution

This class has the following operations:

o+ access pollution data

o+ update current data

o+ update long term data

o+ update pollution data

Violation

This class has the following operations:

os+ create

os+ process violation

8.5 Payment

The class diagram for the key classes of the Payment package is shown in Figure 65.

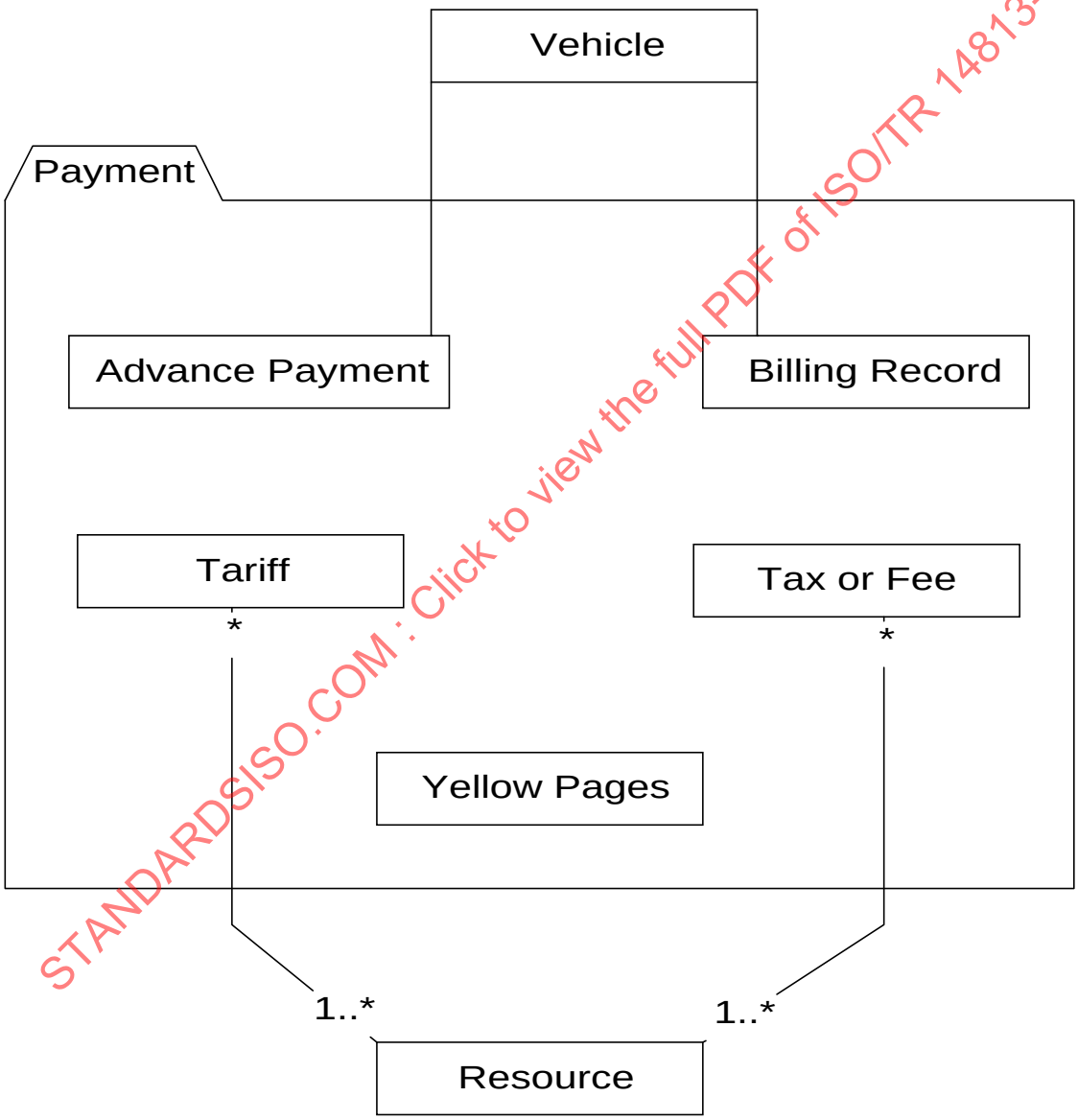

Figure 65 — Key classes of the Payment package

Advance Payment

This class has the following operations:

os+ create

o+ reconcile

Billing Record

This class has the following operations:

os+ create

o+ enter completion details

os+ reconcile collected fees

o+ record creditor amount

os+ select

Tariff

This class has the following operations:

o+ calculate charges

os+ create

o+ request to cater to demand

o+ update tariff

Tax or Fee

This class has the following operations:

o+ access segment charges

Yellow Pages

This class has the following operations:

o+ construct yp service

o+ update general information

o+ update yp service

o+ yp transaction

8.6 Interfaces

The class diagram for the key classes of the interface packages is shown in Figure 66.

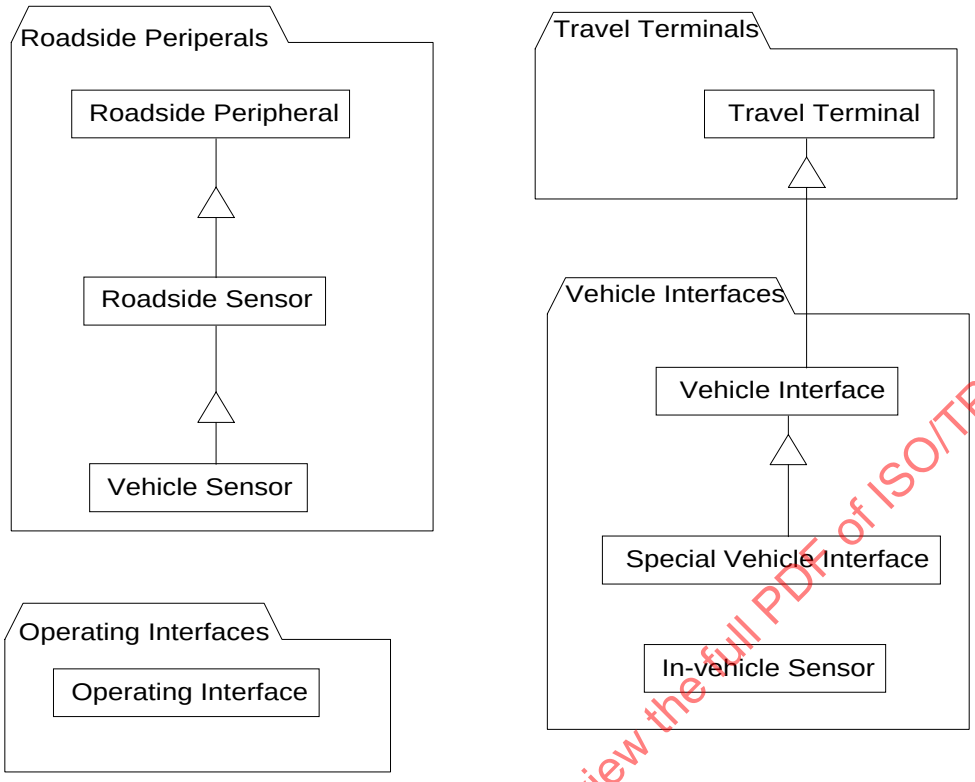

Figure 66 — Key classes of the interface packages

In-vehicle Sensor

This class has the following operations:

- o+ detect passenger embark/disembark
- o+ monitor vehicle and cargo

Operating Interface

This class has the following operations:

- o+ conformance agency check
- o+ freight booking
- o+ instruct operator
- o+ make intermodal reservation
- o+ make parking reservation
- o+ make ridesharing reservation
- o+ notify rail
- o+ operator data output

o+ register user credit card
o+ report violation
o+ request funds transfer
o+ yp service order

Roadside Peripheral

This class has the following operations:
o+ safety warning
o+ update control data

Roadside Sensor

This class has the following operations:
o+ actor in range
Roadside Peripheral

Special Vehicle Interface

This class has the following operations:
o+ access vehicle and passenger data
o+ count passengers
o+ driver action request
o+ interrogate on-board data
o+ track vehicle and status
o+ update driver instructions
Vehicle Interface

Travel Terminal

This class has the following operations:
o+ accept cash/credit
o+ acknowledge
o+ commit to route & schedule
o+ confirm
o+ confirm yp request
o+ debit request
o+ initiate journey request
o+ provide yp advertisement
o+ publish schedule
o+ read payment card/tag

o+ security request

o+ sense security violation

o+ update journey request

o+ yp request

Vehicle Interface

This class has the following operations:

o+ access payment identification

o+ enter route parameters

o+ get location

o+ request operational mode

o+ warn driver

Travel Terminal

Vehicle Sensor

This class has the following operations:

o+ access vehicle identification data

o+ specific actor in range

Roadside Sensor

8.7 Class-Sequence Diagram Matrix

The following clauses detail where classes have been used in a diagram.

8.7.1 Matrix for Key Control and Information Classes

This matrix lists the Class name, followed by Sequence Diagrams (Use Cases).

Advance Payment

Payment Transaction

Billing Record

Fare Collection

Journey Payment

On-trip Traveller Information

Payment Transaction

Vehicle Charges

Carrier

Commercial Vehicle Administrative Processes

Commercial Vehicle
- Commercial Vehicle Administrative Processes
- Commercial Vehicle Road Operation
- Transportation Planning Support

Commercial Vehicle Driver
- Commercial Vehicle Administrative Processes

Commercial Vehicle Tour
- Commercial Vehicle Administrative Processes
- Commercial Vehicle Road Operation
- Incident Management

Control Plan
- Traffic Control
- Transportation Planning Support

Demand Plan
- Demand Management
- Transportation Planning Support

Demand Responsive Vehicle
- Demand Responsive Public Transport

Demand Responsive Vehicle Itinerary
- Demand Responsive Public Transport

Emergency
- Emergency Notification and Personal Security
- Emergency Resources Allocation
- Emergency Vehicle Management
- Incident Management
- Vehicle Status

Emergency Response
- Emergency Resources Allocation
- Emergency Vehicle Management

Emergency Vehicle
- Emergency Resources Allocation
- Emergency Vehicle Management
- Transportation Planning Support

Forwarder
- Order and Shipment

Freight Complement
- Commercial Vehicle Tour Planning

Goods Item
- Commercial Vehicle Tour Planning
- Order and Shipment

Incident
- Incident Management
- Performance Evaluation
- Performance Prediction
- Route Guidance and Navigation
- Traffic Control

Incident Response
- Incident Management
- Transportation Planning Support

Journey
- Commercial Vehicle Administrative Processes
- Commercial Vehicle Tour Planning
- Demand Responsive Public Transport
- Journey Payment
- Journey Schedule
- Order and Shipment
- Pre-journey Information
- Route Guidance and Navigation

Journey Schedule
- Commercial Vehicle Administrative Processes
- Journey Payment
- Journey Schedule

- Pre-journey Information

Local Control Group
- Safety Enhancement for Vulnerable Road Users
- Traffic & Pollution Measurement & Control
- Traffic Control
- Vehicle Status

Movement
- Traffic & Pollution Measurement & Control

O-D Route
- Demand Management
- Performance Evaluation
- Performance Prediction
- Pre-journey Information
- Route Guidance and Navigation
- Route and Schedule Planning
- Transportation Planning Support

Parking
- Demand Management
- Performance Evaluation
- Traffic & Pollution Measurement & Control
- Traffic Control
- Transportation Planning Support

Pollution
- Incident Management
- Performance Evaluation
- Traffic & Pollution Measurement & Control
- Transportation Planning Support

Probe
- Payment Transaction
- Performance Evaluation
- Route Guidance and Navigation
- Transportation Planning Support

Public Transport Route
- Route & Schedule Planning

Public Transport Schedule
- Demand Management
- Fixed Route Public Transport
- Pre-journey Information
- Route & Schedule Planning
- Route Guidance and Navigation
- Transportation Planning Support

Resource
- Commercial Vehicle Road Operation
- Payment Means
- Payment Transaction
- Vehicle Charges

Roadway Group
- Demand Management
- Emergency Resources Allocation
- Emergency Vehicle Management
- Fixed Route Public Transport
- Incident Management
- Infrastructure Maintenance Management
- Performance Evaluation
- Performance Prediction
- Route & Schedule Planning
- Route Guidance and Navigation
- Traffic & Pollution Measurement & Control
- Traffic Control
- Transportation Planning Support

Tariff
- Demand Management
- Payment Transaction

Tax or Fee
- Commercial Vehicle Administrative Processes

Trip

Commercial Vehicle Tour Planning
Journey Schedule
Order and Shipment
Pre-journey Information
Route Guidance and Navigation

Vehicle
Safety Enhancement for Vulnerable Road Users
Vehicle Operation
Vehicle Status

Violation
Commercial Vehicle Road Operation
Payment Transaction
Traffic & Pollution Measurement & Control

Yellow Pages
On-trip Traveller Information

8.7.2 Matrix for Interface Classes

This matrix lists the Class name, followed by Sequence Diagrams (Use Cases).

In-vehicle Sensor
Commercial Vehicle Road Operation
Fare Collection
Safety Enhancement for Vulnerable Road Users
Vehicle Status

Operating Interface
Commercial Vehicle Administrative Processes
Commercial Vehicle Road Operation
Commercial Vehicle Tour Planning
Demand Management
Demand Responsive Public Transport
Emergency Resources Allocation
Emergency Vehicle Management
Fixed Route Public Transport
Incident Management
Infrastructure Maintenance Management

- Journey Schedule
- On-trip Traveller Information
- Order and Shipment
- Payment Means
- Payment Transaction
- Performance Evaluation
- Route & Schedule Planning
- Route Guidance and Navigation
- Traffic & Pollution Measurement & Control
- Traffic Control
- Transportation Planning Support

Roadside Peripheral
- Safety Enhancement for Vulnerable Road Users
- Traffic & Pollution Measurement & Control
- Traffic Control

Roadside Sensor
- Traffic & Pollution Measurement & Control

Special Vehicle Interface
- Commercial Vehicle Road Operation
- Demand Responsive Public Transport
- Emergency Resources Allocation
- Emergency Vehicle Management
- Fare Collection
- Fixed Route Public Transport
- Route & Schedule Planning
- Traffic Control

Travel Terminal
- Demand Responsive Public Transport
- Emergency Notification and Personal Security
- Emergency Resources Allocation
- Fare Collection
- Fixed Route Public Transport
- Journey Payment
- On-trip Traveller Information
- Order and Shipment
- Payment Means

- Payment Transaction
- Pre-journey Information
- Route & Schedule Planning
- Route Guidance and Navigation

Vehicle Interface
- On-trip Traveller Information
- Route Guidance and Navigation
- Safety Enhancement for Vulnerable Road Users
- Traffic & Pollution Measurement & Control
- Vehicle Charges
- Vehicle Operation
- Vehicle Status

Vehicle Sensor
- Commercial Vehicle Road Operation
- Traffic & Pollution Measurement & Control
- Vehicle Charges

9 Package Interfaces

The control and information classes defined in 8.1 to 8.5 interact with the actors through interface classes identified in 8.6. The collaboration between the interface classes and the other classes is documented in the following clauses.

Each clause contains a diagram with a collaboration linkage shown between each class for which there is a corresponding message in the sequence diagrams of clause 7. There may be multiple messages between each class pair. The direction of the message from the originator is shown by arrow and the message is identified by a label of the form

Sequence Diagram:operation name

where the class of the operation can be derived by following the arrow direction on the diagram. For example, in Figure 67, there is a message from the interface class Special Vehicle Interface to the Roadway Group class:

Emergency Vehicle Management:priority request(...).

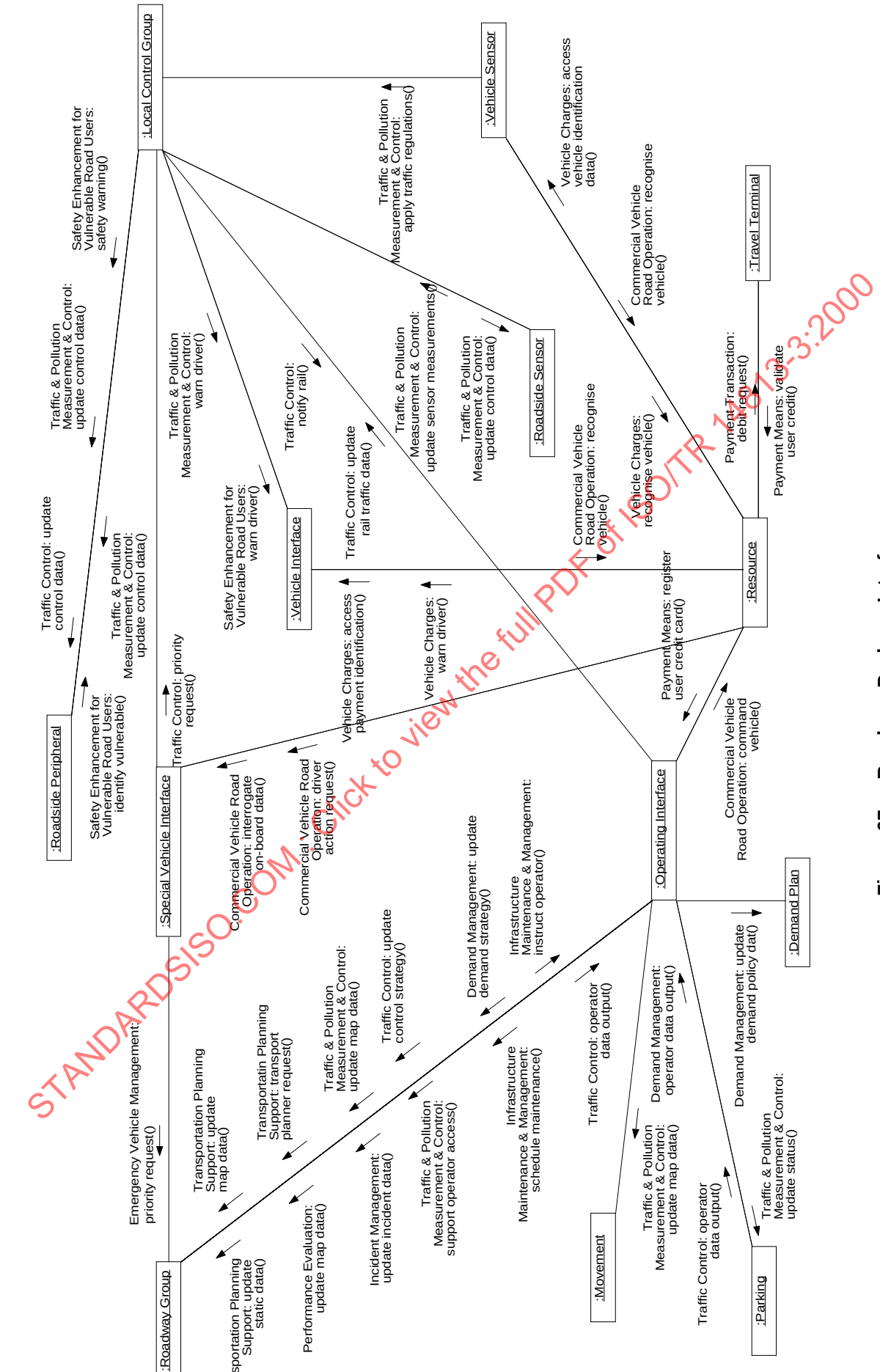

Figure 67 — Roadway Package Interfaces

9.1 Roadway Interfaces

The collaboration between the Roadway classes and the Interface classes is documented in Figure 67. The number of message types between collaborating pairs is summarised in Table 10.

Table 10 — Matrix count of Roadway class Interface class collaborations

	Roadway Group	Local Control Group	Resource	Demand Plan	Movement	Parking
Operating Interface	12	2	2	1	1	3
Vehicle Interface		2	3			
Special Vehicle Interface	1	1	2			
Roadside Peripheral		4				
Roadside Sensor		2				
Vehicle Sensor		1	3			
Travel Terminal			2			

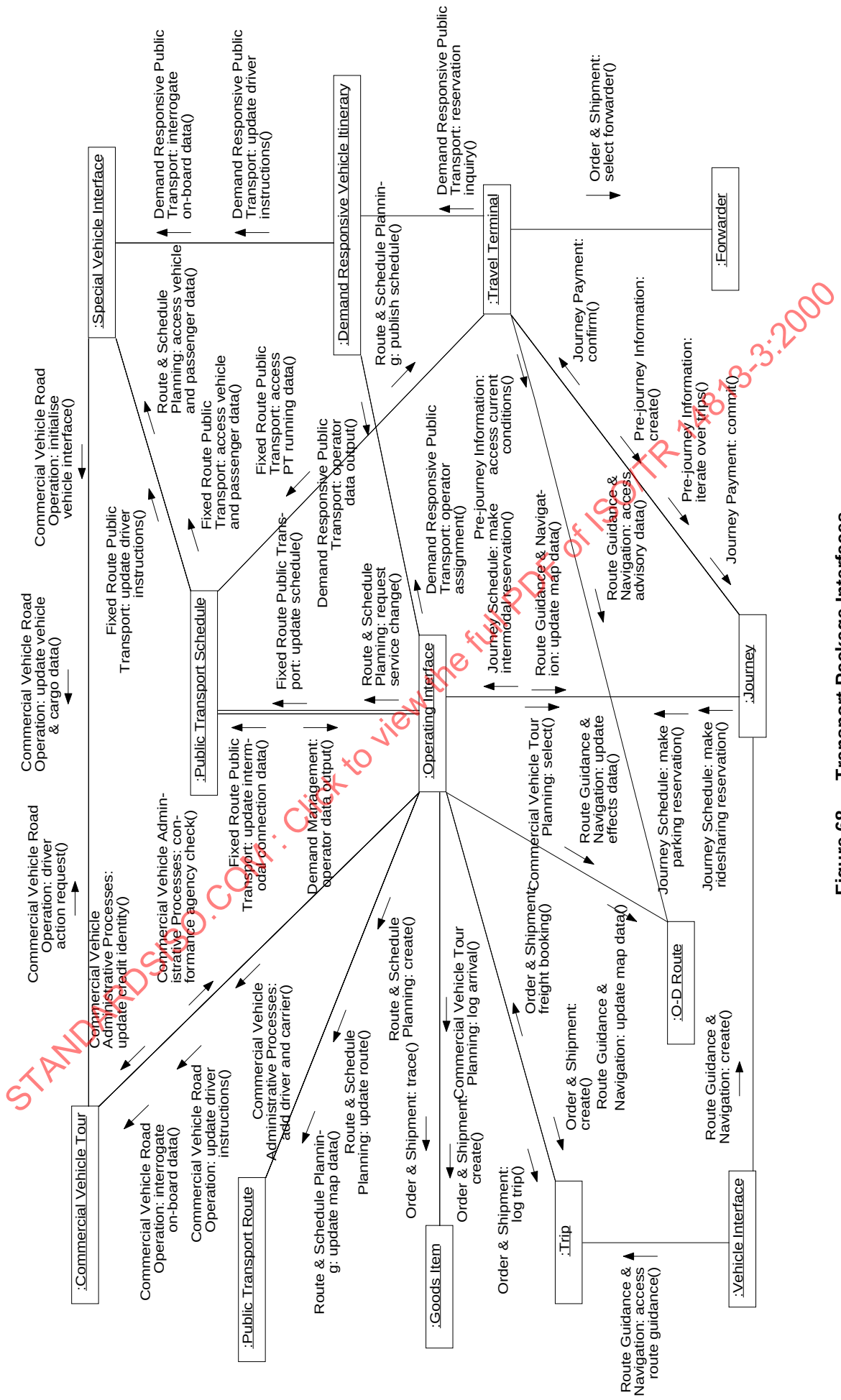

Figure 68 — Transport Package Interfaces

9.2 Transport Interfaces

The collaboration between the Transport classes and the Interface classes is documented in Figure 68. The number of message types between collaborating pairs is summarised in Table 11.

Table 11 — Matrix count of Transport class Interface class collaborations

	Operating Interface	Vehicle Interface	Special Vehicle Interface	Travel Terminal
Commercial Vehicle Tour	5		3	
Public Transport Route	3			
Goods Item	3			
Trip	3	1		
O-D Route	2			2
Journey	5	1		4
Forwarder				1
Demand Responsive Vehicle Itinerary	2		2	1
Public Transport Schedule	4		3	2

9.3 Vehicle Interfaces

The collaboration between the Vehicles classes and the Interface classes is documented in Figure 69. The number of message types between collaborating pairs is summarised in Table 12.

Table 12 — Matrix count of Vehicles class Interface class collaborations

	In-vehicle Sensor	Vehicle Interface
Vehicle	5	6

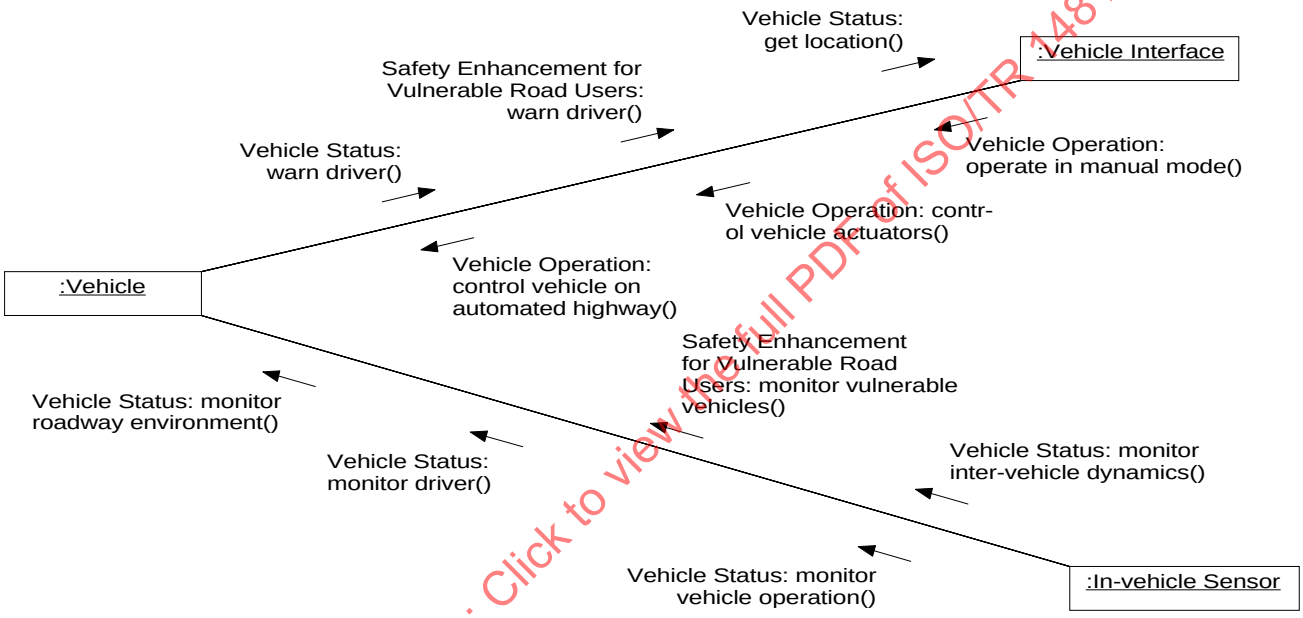

Figure 69 — Vehicle Package Interfaces

9.4 Events Interfaces

The collaboration between the Events classes and the Interface classes is documented in Figure 70. The number of message types between collaborating pairs is summarised in Table 13.

Table 13 — Matrix count of Events class Interface class collaborations

	Operating Interface	Special Vehicle Interface	Roadside Sensor	Travel Terminal
Incident Response	1			
Incident				
Emergency Response	1		3	
Emergency	4	2		4
Violation	3			
Pollution			1	

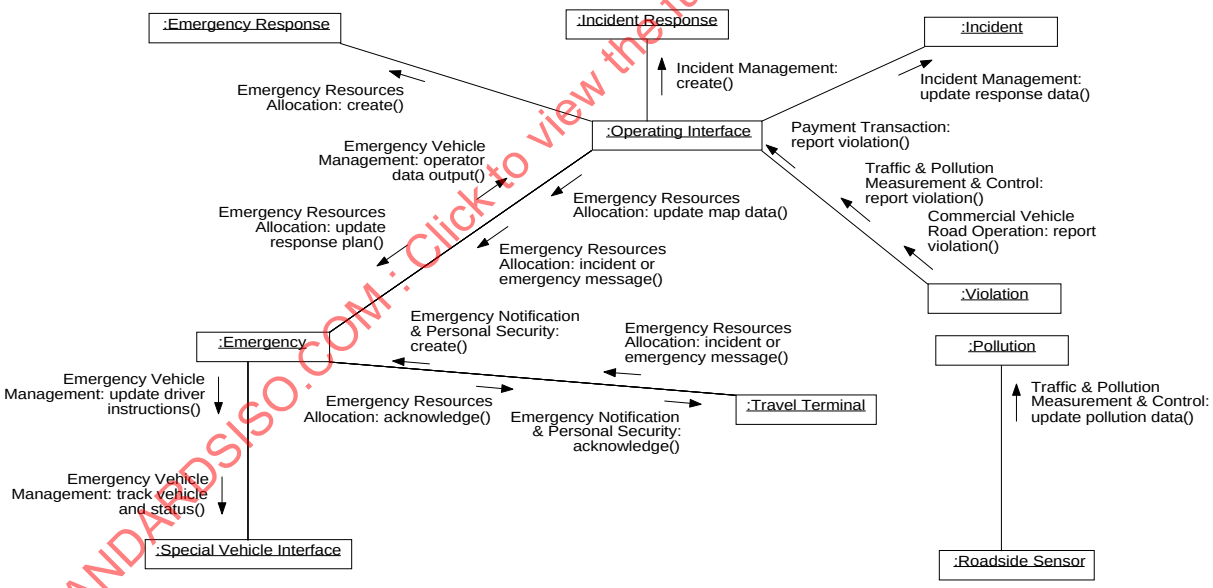

Figure 70 — Events Package Interfaces

9.5 Payment Interfaces

The collaboration between the Payment classes and the Interface classes is documented in Figure 71. The number of message types between collaborating pairs is summarised in Table 14.

Table 14 — Matrix count of Payment class Interface class collaborations

	Operating Interface	Travel Terminal
Yellow Pages	4	3
Billing Record	1	1
Tariff	2	

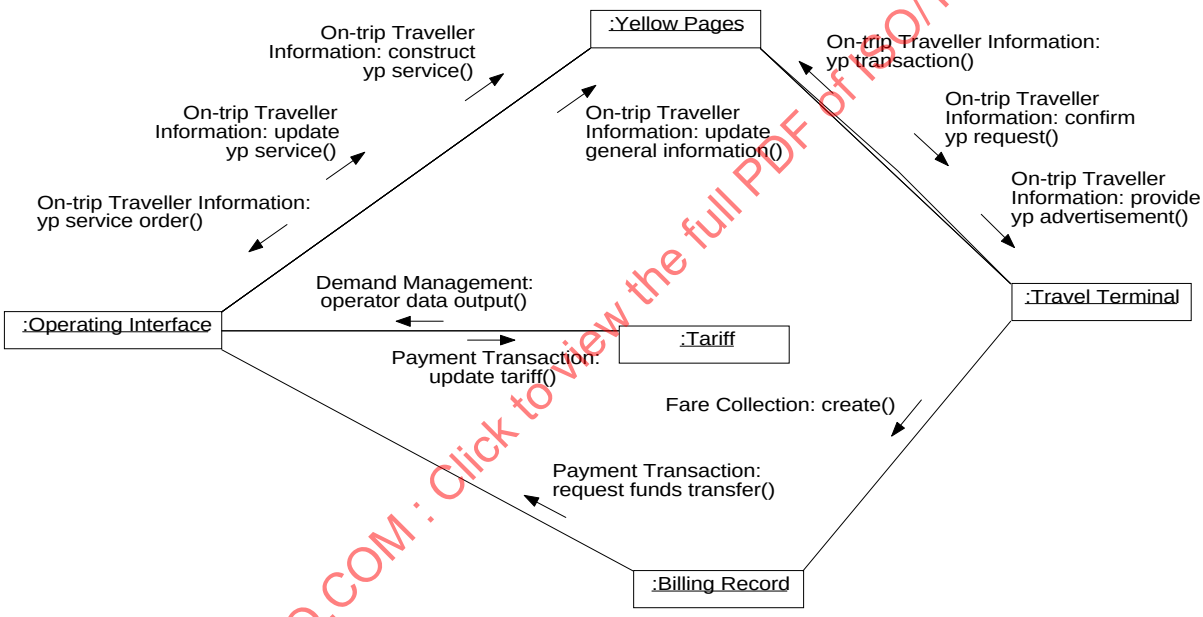

Figure 71 — Payment Package Interfaces

10 Dependencies between Packages

The control and information classes defined in 8.1 to 8.5 interact as shown in the sequence diagrams of Clause 7. The collaboration between the classes of the various packages is documented in the following clauses.

Each clause contains a diagram with a collaboration linkage shown between each pair of classes in different packages for which there is a corresponding message in the sequence diagrams of clause 7. There may be multiple messages between each class pair. The direction of the message from the originator is shown by arrow and the message is identified by a label of the form

Sequence Diagram:operation name

where the class of the operation can be derived by following the arrow direction on the diagram. For example, in Figure 72, there is a message from the class O-D Route of the Transport package to the Probe class of the Roadway package:

Route Guidance & Navigation:access route travel time(...).

10.1 Roadway - Transport Collaboration

The collaboration between the Roadway classes and the Transport classes is documented in Figure 72. The number of message types between collaborating pairs is summarised in Table 15.

Table 15 — Matrix count of Roadway class Transport class collaborations

	Roadway Group	Resource	Probe
O-D Route	8		1
Commercial Vehicle Tour	1	1	
Emergency Vehicle	1		
Commercial Vehicle	1	1	
Public Transport Schedule	6		

10.2 Roadway - Vehicles Collaboration

The collaboration between the Roadway classes and the Vehicles classes is documented in Figure 73.

10.3 Roadway - Events Collaboration

The collaboration between the Roadway classes and the Events classes is documented in Figure 74. The number of message types between collaborating pairs is summarised in Table 16.

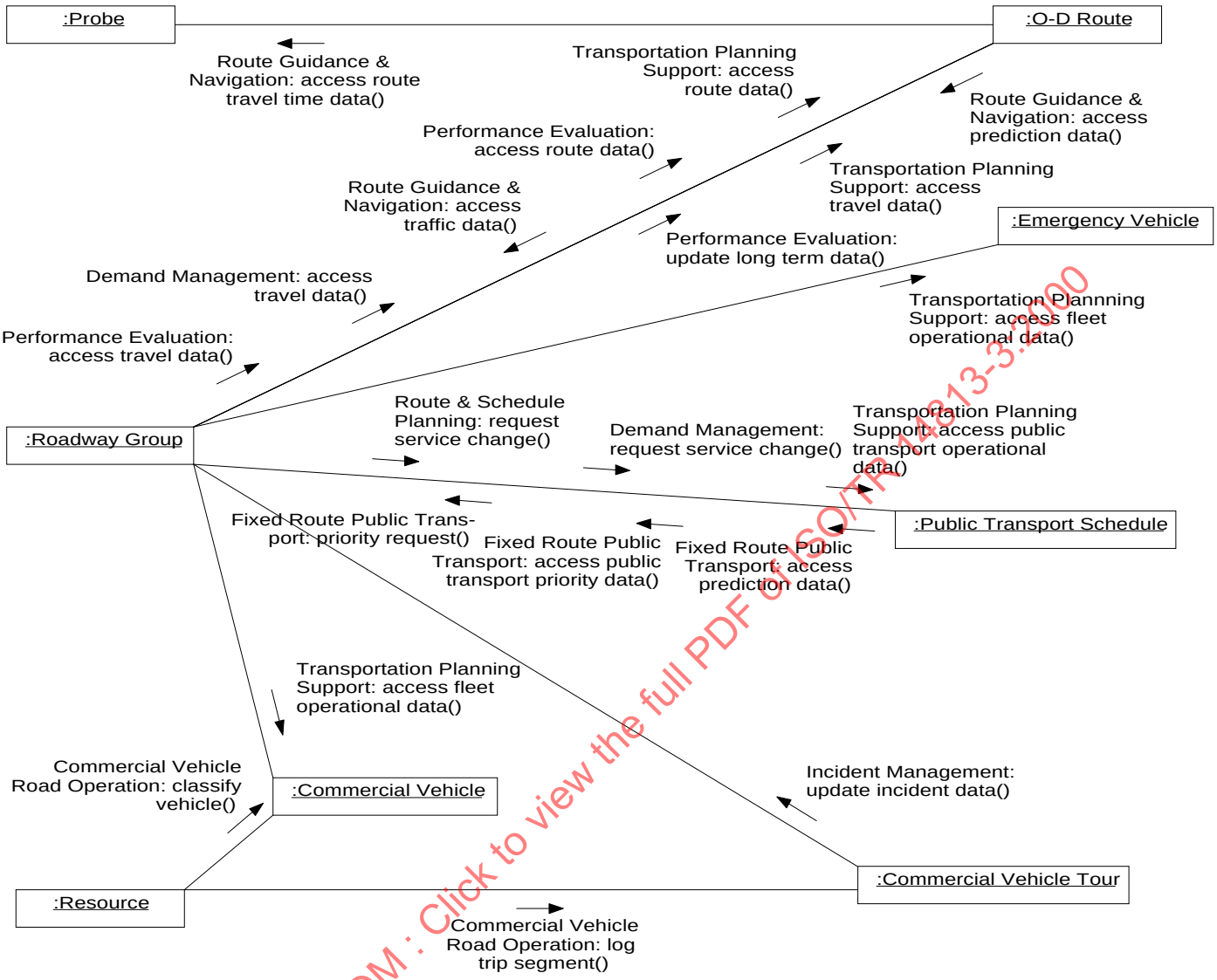

Figure 72 — Roadway - Transport Collaborations

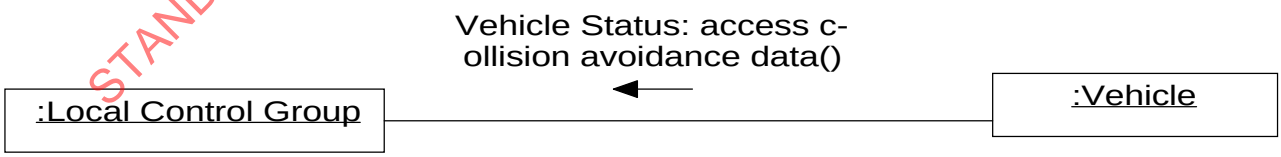

Figure 73 — Roadway - Vehicles Collaborations

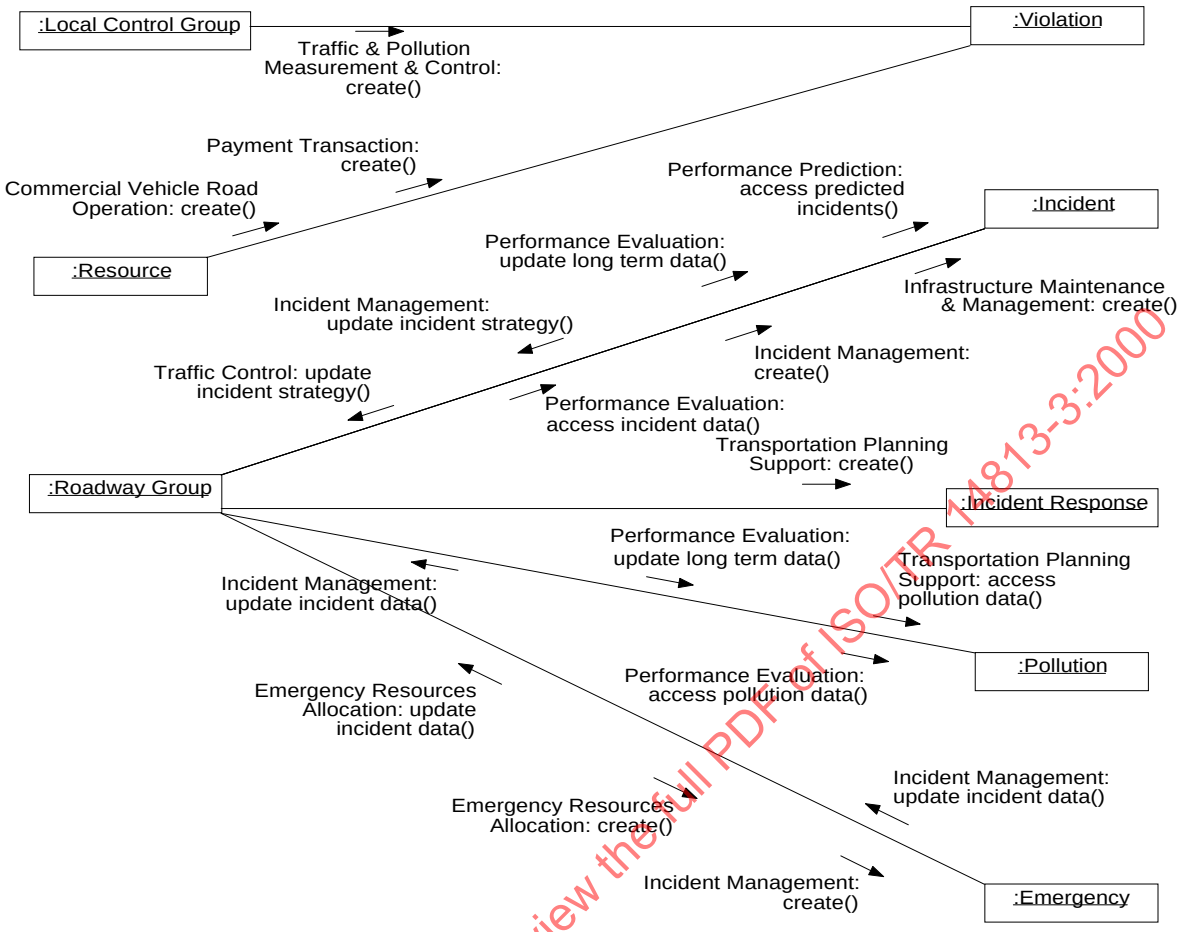

Figure 74 — Roadway - Events Collaborations

Table 16 — Matrix count of Roadway class Events class collaborations

	Local Control Group	Resource	Roadway Group
Violation	1	2	1
Incident			7
Incident Response			1
Pollution			4
Emergency			4

10.4 Roadway - Payment Collaboration

The collaboration between the Roadway classes and the Payment classes is documented in Figure 75. The number of message types between collaborating pairs is summarised in Table 17.

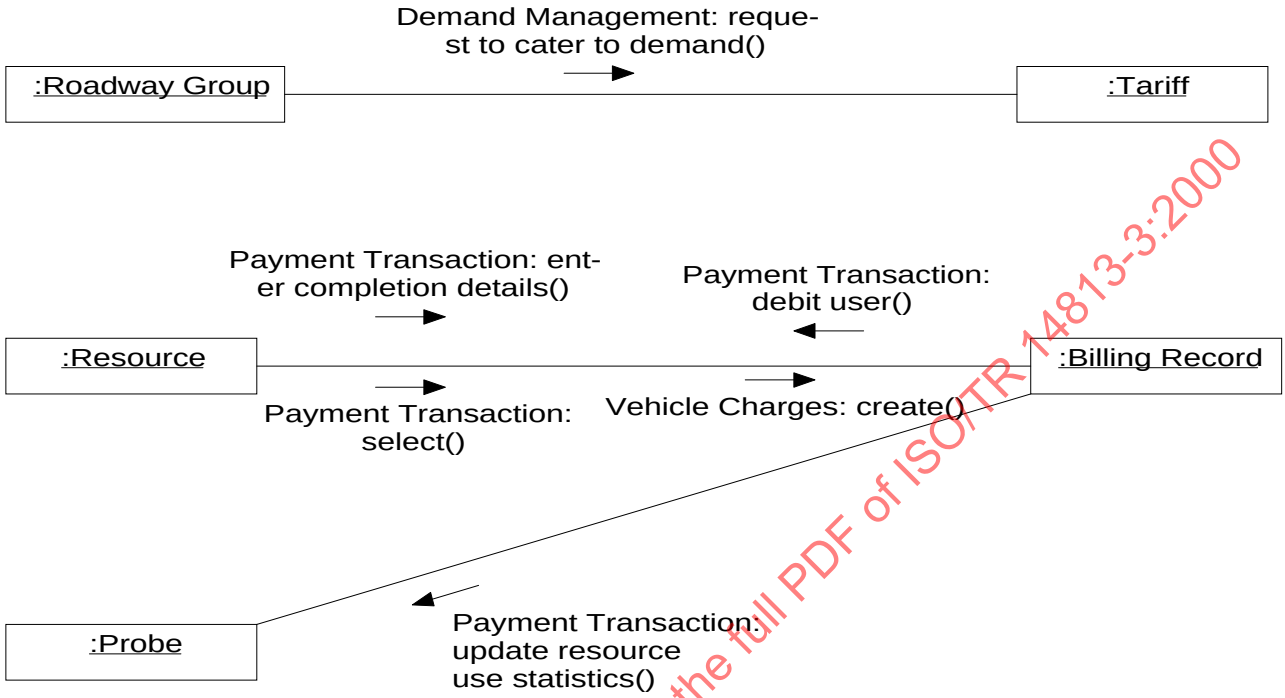

Figure 75 — Roadway - Payment Collaborations

Table 17 — Matrix count of Roadway class Payment class collaborations

	Probe	Resource	Roadway Group
Tariff			1
Billing Record		4	1

10.5 Transport - Events Collaboration

The collaboration between the Transport classes and the Events classes is documented in Figure 76.

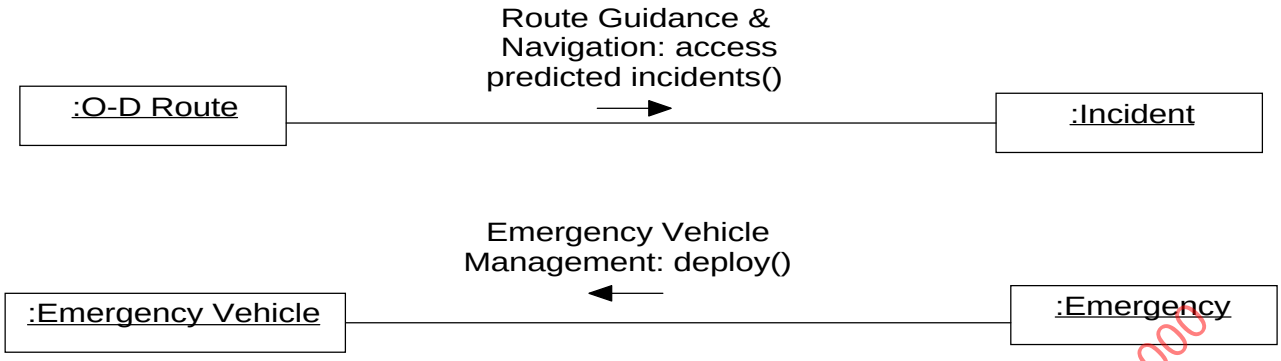

Figure 76 — Transport - Events Collaborations

10.6 Transport - Payment Collaboration

The collaboration between the Transport classes and the Payment classes is documented in Figure 77.

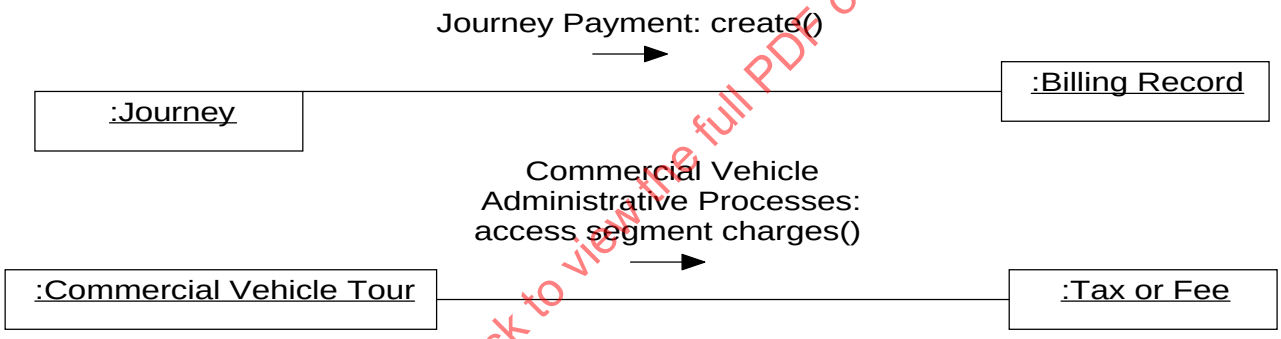

Figure 77 — Transport - Payment Collaborations

10.7 Vehicles - Events Collaboration

The collaboration between the Vehicles classes and the Events classes is documented in Figure 78.

Figure 78 — Vehicles - Events Collaborations

Annex A
(informative)

Management and Information Centres Information Classes

The draft standard ISO 14827 **Message Format and Information Contexts for Traffic Management and Information Centres (TMIC)** has been used to derive an aggregation hierarchy of information classes used in classes and sequence diagrams pertinent to Traffic Management and Traveller Information classes. A data dictionary for these classes is given in this Annex. It is divided into nine clauses according to the tree defined in Figure A.1.

ISO 14827 is under revision by TC204/WG9 and the definitions given here are based on draft N173H, (1998-01-08). The descriptions of the data elements provided in N173H have been copied verbatim into this dictionary, except that the description is usually introduced with the words “This information class shall consists of...”. The terminology used in N173H is not consistent with the object-oriented model and so terms such as “file” and “list” occur. The reader will have to translate these terms into appropriate object-oriented terms, (e.g. file and list might correspond to the members of a class, or to the attributes of a complex object).

In Figure A.1 the class [0000] TMIC is a virtual base class to define global attributes.

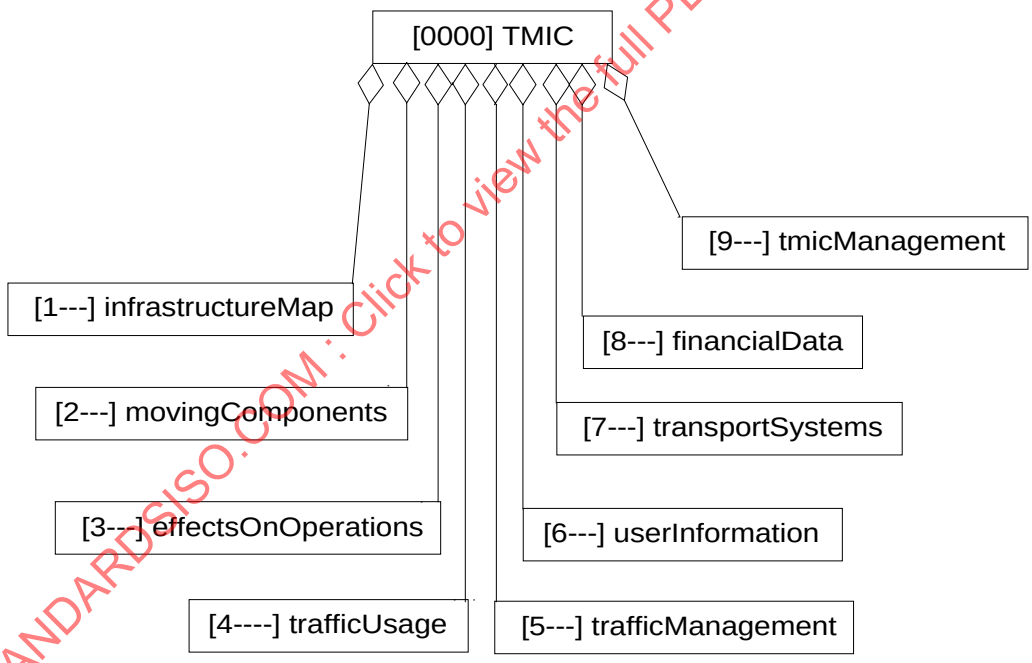

Figure A.1 — The top level of the naming tree for the information classes defined in the data dictionary

A.1 Infrastructure Map

[1---] infrastructureMap

These are maps of the static physical network. The units used at this level may be specified in DC[13].

Attributes:

name/address [[1---] infrastructureMap.] The name/address of the group responsible for the information class.

[11--] generalMap This aggregate information class shall consist of a layered map of a region, including its transport system. This base map describes all relevant geographic features of the region. It is desirably a digital, GIS - layered map in GDF format based on the work of TC204/WG3. The version (or revision or edition) is covered by the obligatory DC[14b]. All the map layers described elsewhere in [1---] are uniquely linked to the map. The contents of the map should be acceptable to all the traffic and transport operators using the system. The map can be formally authenticated by users, employing the DC[15] facility. The map publisher (or assembler or producer) is uniquely specified by mandatory attribute A1, and each layer publisher by attribute A2.

Attributes:

map name/ID [[11--] generalMap.] Identifying

region name/ID [[11--] generalMap.] Identifying

[111-] layer The map layer publisher (or assembler or producer) is uniquely specified by mandatory attribute.

Attributes:

layerID [[111-] layer.] Administrative.

publisher [[111-] layer.] Administrative.

[12--] transportMap These are facilities operated by the relevant service provider.

[121-] ptRoute This information class shall consist of a map layer of pt routes and stations. This is a map layer of pt routes, bus stops, train stations, access links, facilities for the handicapped, etc., with particular emphasis on road-based routes. pt schedules are covered in [721-].

Attributes:

ease of use of station [[121-] ptRoute.] Definitional Attribute.

fare structure [[121-] ptRoute.] Definitional Attribute.

link travel time [[121-] ptRoute.] Definitional attribute. Should be considered for both light and congested conditions and for both normal and handicapped travellers.

time for passenger to traverse station [[121-] ptRoute.] Definitional attribute. The transfer process associated with Definitional attribute is described in [75--].

transit type [[121-] ptRoute.] Identifying attribute. Will cover whether heavy rail, light rail, bus, etc.

[122-] transportNode

[1221] parkingFacility This information class shall consist of a map layer of parking facilities. This is a map layer locating car parks and parking stations. The supply of available spaces is covered in [7131].

Attributes:

usage fees [[1221] parkingFacility.] Definitional attribute.

[1222] freightTerminal This information class shall consist of a map layer of freight terminals. The supply of available spaces is covered in [7133].

Attributes:

usage fees [[1222] freightTerminal.] Definitional attribute.

[1223] rideSharePickup This information class shall conist of a map layer of ride-share pick-up points. This is a map layer of nominated pick-up points in an endorsed ride-share (car-pooling) system.

Attributes:

organiser's name/ID [[1223] rideSharePickup.] Identifying attribute.

ride-share rules [[1223] rideSharePickup.] Definitional attribute.

[1224] ptStation This information class shall consist of a map layer of pt stations. This map layer will often be incorporated into [121-]. Station includes any pt passenger access point.

Attributes:

station ID [[1224] ptStation.] Identifying 1

station access parameters [[1224] ptStation.] Definitional attribute

[13--] roadMap

[131-] travelledWay Road (or link, although link often involves direction) includes bridges, ferries, and tunnels. Routine operating characteristics are covered in [132-], as all roads are considered to be comprised of operating lanes.Short-term closures would be initially noted in [132-]. However, [1311] will subsequently be updated if an entire road closes, is blocked, becomes impassable, or reopens.

Attributes:

chainage [[131-] travelledWay.]

classification/type [[131-] travelledWay.] Identifying 2

name/ID [[131-] travelledWay.] Identifying 1

[1311] roadLayer This information class shall consist of a map layer of the road network, its links and nodes, and their connectivity. This layer gives the location of the road network. It also gives the distance (chainage) along the road, from some defined but arbitrary datum, which is used often used to locate road features.

Attributes:

link average travel time [[1311] roadLayer.] Definitional attribute.

[1312] pathLayer This information class shall consist of a map layer of independent paths. This layer gives the independent paths, usually restricted to pedestrians, bicycles and similar vehicles. Various jurisdictions will use attribute Definitional attribute differently.

Attributes:

cross-section [[1312] pathLayer.] Definitional attribute.

surfacing [[1312] pathLayer.] Definitional attribute.

user restrictions [[1312] pathLayer.] Definitional attribute.